
SECTION 8

OPERATOR INTERFACE*

J. N. Beach

*MICRO SWITCH Division, Honeywell Inc., Freeport, Illinois
(Operator Interface—Design Rationale)*

Kai-Chun Cheng

*School of Industrial Engineering, Purdue University, West
Lafayette, Indiana (Cognitive Skills and Process Control)*

Ray E. Eberts

*School of Industrial Engineering, Purdue University, West
Lafayette, Indiana (Adaptation of Manuscript on “Cognitive Skills
and Process Control)*

Gregory K. McMillan

*Solutia Inc., St. Louis, Missouri (Knowledge-Based Operator
Training, Intelligent Alarms)*

James A. Odom

*Corporate Industrial Design, Honeywell Inc., Minneapolis,
Minnesota (Operator Interface—Design Rationale)*

Staff

*Moore Products Company, Spring House, Pennsylvania
(CRT-Based Graphic Display Diagrams)*

OPERATOR INTERFACE—DESIGN RATIONALE	8.2
HUMAN FACTORS	8.3
Importance of Initial Planning	8.3
Operator Variables	8.3
Habit Patterns	8.3
Operator-Interface Geometry	8.4
Alphanumeric Displays	8.5
Keyboards	8.8
Voice Recognition	8.10
ENVIRONMENTAL FACTORS	8.10
Local Environment	8.10
Ambient Light Conditions	8.11
COGNITIVE SKILLS AND PROCESS CONTROL	8.11
INTRODUCTION	8.11
PROCESS CONTROL TASKS	8.11
MODELS OF PROCESS CONTROL TASKS	8.12

* Persons who authored complete articles or subsections of articles, or otherwise cooperated in an outstanding manner in furnishing information and helpful counsel to the editorial staff.

Quantitative Models	8.12
GOMS and NGOMSL	8.12
Expert Systems	8.13
Conclusions	8.13
EXTRACTING EXPERT KNOWLEDGE	8.14
Protocol Analysis	8.14
Scaling	8.14
Pattern Recognition	8.14
EXPERTISE IN PROCESS CONTROL	8.15
Automatization	8.15
Accurate Mental Models	8.15
Spatial Representations	8.15
TRAINING COGNITIVE SKILLS	8.16
Using a Predictor	8.16
Spatial Information	8.17
Analogies and Metaphors	8.17
Active Learning	8.17
Learning Styles	8.17
IMPACT OF AUTOMATION	8.17
Vigilance Decrement	8.18
Out-of-Loop Familiarity	8.18
Overestimating the Intelligence of the Computers	8.18
Information Overload	8.19
CONCLUSIONS	8.19
REFERENCES	8.20
DISTRIBUTED DISPLAY ARCHITECTURE	8.22
THE GRAPHICS INTERFACE	8.22
USER INTERACTIONS	8.25
“CONVERSATIONAL” INTERACTIONS	8.25
DATA INPUTS	8.26
VISUAL DISPLAY DEVICES	8.29
KNOWLEDGE-BASED OPERATOR TRAINING	8.31
REFERENCES	8.36
INTELLIGENT ALARMS	8.37
REFERENCES	8.39

OPERATOR INTERFACE—DESIGN RATIONALE*

The interface between a process or machine and the operator is the primary means for providing dialogue and for introducing human judgment into an otherwise automatic system. Although signals arrive at the interface and, once an operator judgment is made, leave the interface at electronic speeds, the operator responds at a much slower communication rate. Thus the operator, because of human limitations, is a major bottleneck in the overall system. The interface, whether it takes the form of

* Technical information furnished by *MICRO SWITCH*, a Division of Honeywell Inc., and James A. Odom, Corporate Industrial Design, Honeywell Inc., Minneapolis, Minnesota.

a console, a workstation, or other configuration, must be designed with the principal objective of shortening operator response time. The interface must be customized to the operator, and through a serious training program, the operator must become accustomed to the interface. The interface designer not only considers the hardware interface, but the software interface as well.

Inadequate interface design usually is a result of (1) considering the interface late in the overall system design process, (2) giving short shrift to human factors, and (3) procuring off-the-shelf interface configurations that have been compromised for generalized application rather than for specific needs. Cost cutting at the interface level of a system carries large risks of later problems and dissatisfaction.

HUMAN FACTORS

Interface design falls within the realm of human factors, which is sometimes referred to as human engineering or ergonomics—all of which pertain to the very specialized technology of designing products for efficient use by people. Human factors is concerned with *everything* from specific characteristics of interface components to the total working environment of the operator.

Importance of Initial Planning

An excellent starting point for the interface designer is that of providing a functional description of the interface and then a job description for the operator. These two descriptions should dovetail precisely. A pro forma questionnaire can be helpful (Fig. 1).

Operator Variables

The principal characteristics of the operator in designing an interface are (1) physical parameters, (2) experience, including trainability, and (3) long-established habit patterns. The physical aspects will be described shortly. In terms of experience, the amount of instruction required for efficiently using the interface as intended obviously depends on the complexity of the process or machine under control and of the interface per se. There are tremendous differences, ranging from the simplicity of operating an average copy machine to a complex machine tool or assembly line to a complex chemical process that incorporates many hundreds of control loops. When forecasting the amount of instruction that an operator will require for a given interface, the designer should establish the specific content of a training program as part of the overall interface design task.

Habit Patterns

People, as the result of past exposure, “expect” controls to move in certain ways. These expectations sometimes are called population stereotypes because they are so universally accepted. Where possible, component selection for an industrial control interface should be an extension of these stereotypes or habit patterns. For example, the wall-mounted toggle switch found in houses has established a habit pattern for turning on the lights. The upward flipping motion is associated with “on” and can be utilized with other instrumentation-type toggle-paddle switches for a natural transfer of a previously learned habit.

The clockwise motion of a rotary knob is frequently used to turn on a domestic appliance (television, range or oven, mixer). This same familiar action may be adapted to any control panel for an extension of a normal habit pattern. The scale of a slide switch or potentiometer should show an increase as the switch is moved upward or to the right. These control actions require the least amount of conscious effort to learn and are well established in our daily lives (Fig. 2).

Initial Planning Sheet—Interface Design	
1. Functionality of interface:	
Variables to be indicated, recorded, adjusted, automatically alarmed	
Provide instructions, ancillary and historical information—Printed or computer-stored	
Extent of interactive graphics desired	
Special tasks of interface	
Will interface satisfy only functional needs, or will it be used also as a “show piece” for VIP visitations?	
2. What are cost constraints? space limitations?	
3. Are there special environmental considerations?—Installed indoors (air-conditioned space?), outdoors, proximity to excessive noise, near moving machinery, vibration and shock, electrical interference, corrosive or explosive vapors	
Will design require guards, barriers or protective shields for components and user safety?	
What degree of “ruggedness” is required?	
4. Is there a requirement for information confidentiality?	
Will interface be used constantly (three shifts) or be out of action parts of a 24-hour period or weekends?	
5. Will operator instructions be simple or complex?	
What will be the extent of operator training?	
6. Who will install equipment and maintain it?—User technicians, supplier, outside service contractor?	
How can installation and maintenance tasks be simplified?	
7. What are the choices in terms of overall physical configuration?—Console, workstation, pedestal, desk?	
8. What are hardware options?—Visual displays, audio alarms, voice reception and activation	
Operator’s physical contact with interface, switches, touch screens, etc.?	
9. Are there any special procurement regulations, as for military installations?	
10. Must the interface fit well into facility expansion plans?	
11. What are the physical and mental profiles of your present very successful operators?	
12. What have been common operator complaints in the past?	
13. Will complexity justify preparing a life-size mockup for achieving an interface that combines human factors with aesthetics?	
14. Functionality of operators—Complete job description, operator qualifications, physical size, agility, alertness, other factors (from past experience)	
Is interface for a process or machine that could cause loss of life or equipment if permitted to go out of control?	

FIGURE 1 Pro forma questionnaire for use by engineers and designers when initially considering new or revised operator interface.

When controls or control and display arrangements take advantage of these habit patterns, generally the following results can be expected:

1. Reaction time is reduced.
2. The first control movement by an operator is usually correct.
3. An operator can perform faster and can make adjustments with greater precision.
4. An operator can learn control procedures faster.

Operator-Interface Geometry

Extensive use of computer data terminals has generated a wide variety of interface architectures. However, there are certain common denominators. Typically, an operator will use a keyboard, a display unit, and a document station (a work holder for printed material). These elements require careful attention to their positioning with respect to the operator. A thorough ergonomic analysis

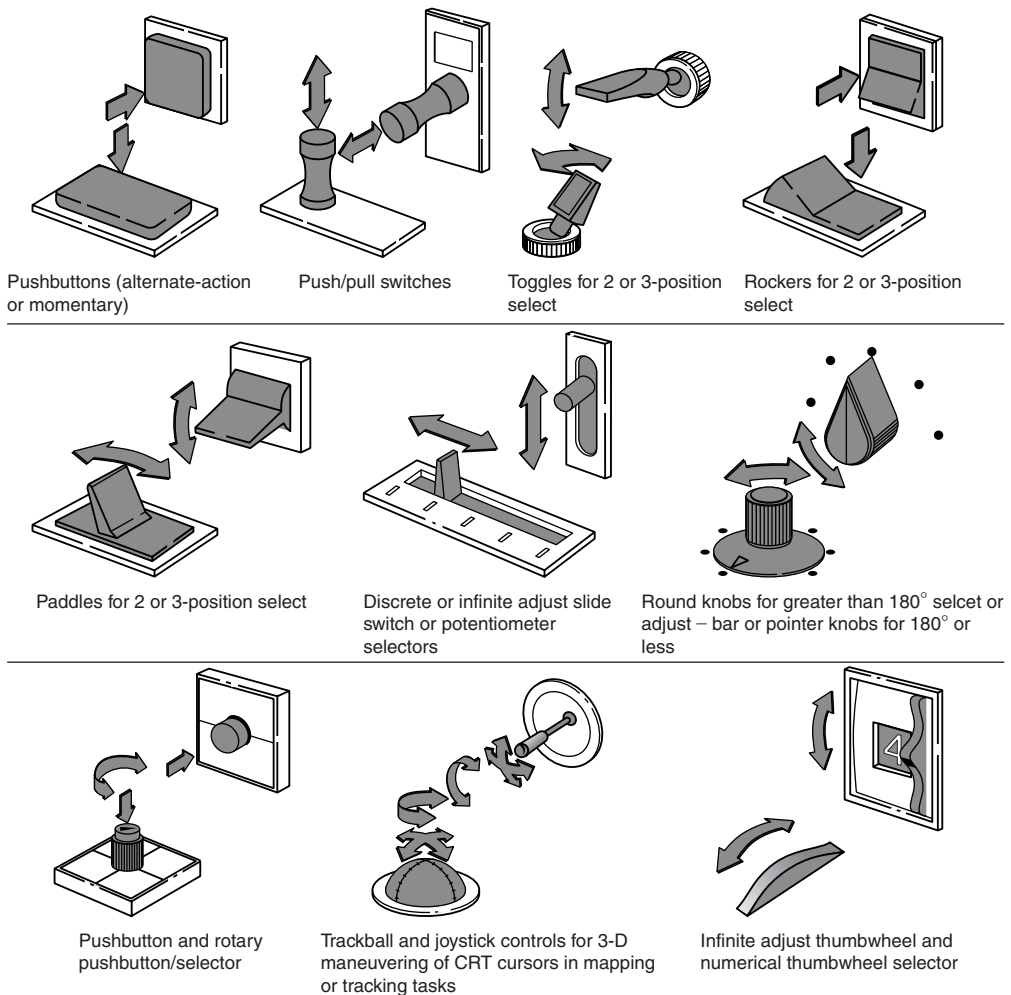


FIGURE 2 Switch response control movements in accordance with habit patterns. (MICRO SWITCH.)

must be a part of interface design to ensure that lines of sight, reach, lighting, access, and angular relationships are addressed properly (Fig. 3).

The line of sight from the operator to the center of a display should be as near to perpendicular as possible. The top of the display should be at or below eye level.

Height, tilt, and swivel adjustment of seat, keyboard, display, supports, and footrests must be provided, since they are all parts of an interactive workplace system. Adjustments for operator posture should be easy to make and without requiring special tools, skills, or strength. If an operator cannot maintain a comfortable posture, there will be complaints of fatigue related to the eyes, neck, shoulders, arms, hands, trunk, legs, and feet.

Alphanumeric Displays

Data display terminals that use cathode-ray tubes (CRTs) are commonly referred to as video display terminals (VDTs) or video display units (VDUs). Data display systems also may use gas discharge,

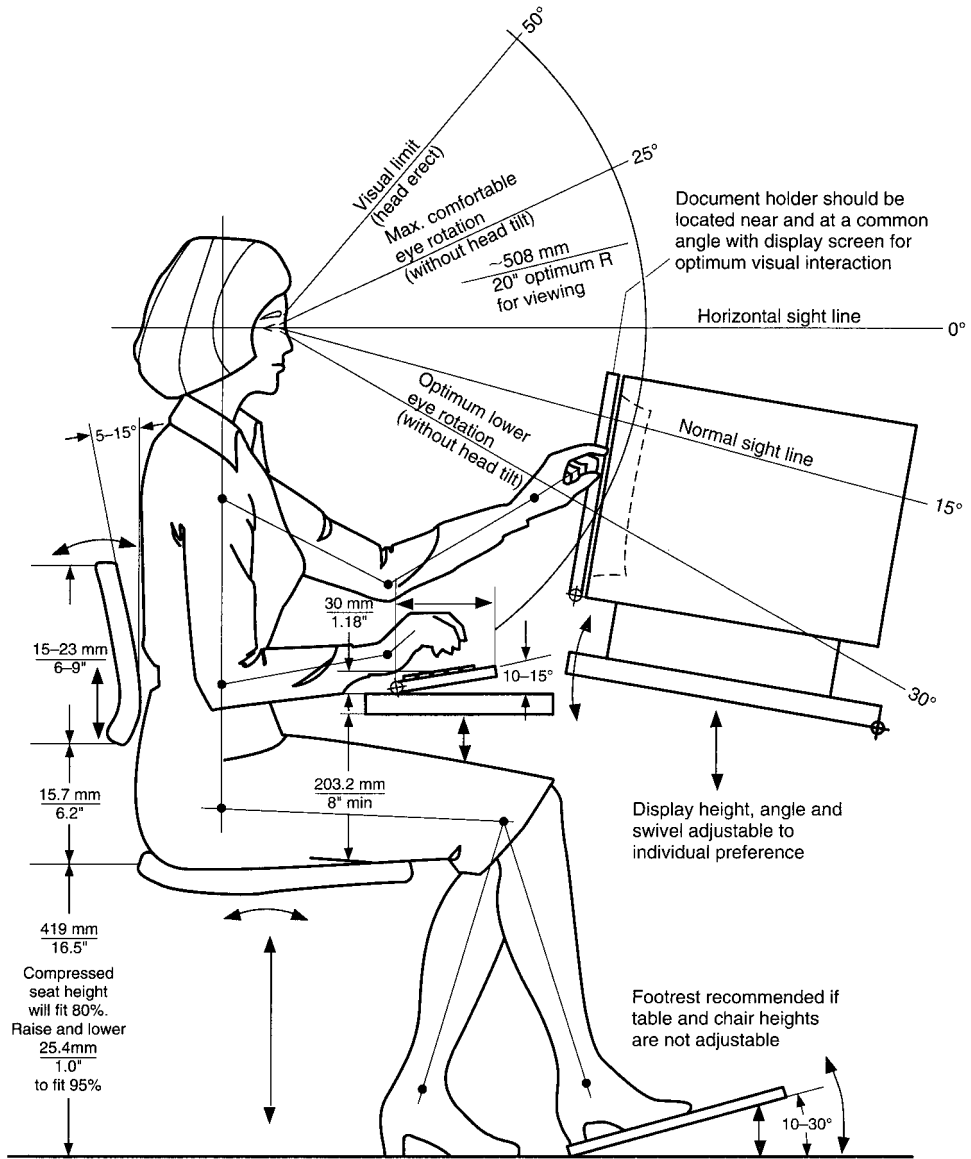


FIGURE 3 Geometry of operator-interface design. (MICRO SWITCH.)

light-emitting diodes (LEDs), and liquid-crystal displays (LCDs). Although non-CRT systems are often attractive because the interface screen can be relatively thin [2.5 to 5 cm (1 to 2 inches) front to back], the CRT enjoys wide usage because of its versatility in terms of graphics and use of color, and because of cost.

Although it is possible to display alphanumeric data on a single or abbreviated line display, as offered by some electronic typewriters, the full-page business-letter format usually is more desirable. If reference material is in the same format as the displayed information, the visual interaction between the two is compatible and operator perception problems are minimized. Many word-processing display



FIGURE 4 Types of alphanumeric matrix displays. (Top) 5 by 7 dot matrix. (Middle) 7 by 9 dot matrix. (Bottom) Bar matrix. (MICRO SWITCH.)

screens offered today do not have full vertical page capacity, but they move type upward (scrolling) when the lower line limit has been reached.

Alphanumeric displays on VDTs are typically dot-matrix construction because of the discrete addressing mode which they use. Both 5 by 7 and 7 by 9 rectangular dot groupings are used, with the larger sides vertical (Fig. 4). A practical working height for these characters is 2.5 mm (0.1 inch) minimum. With adequate spacing between characters and lines, a typical display screen [244 mm wide by 183 mm high (9.6 by 7.2 inches)] can reasonably accommodate 47 lines of 95 characters.

Although light characters on a dark background are most common in VDTs, testing has verified that there is improved legibility with dark characters on light backgrounds. By having the same contrast format for both reference document and display, there is considerably less eye strain as an operator shifts back and forth between two surfaces. It is good practice to have CRT displays include the capability of image reversal from positive to negative in full or selected areas. Dark characters on a CRT light background display (positive image) present less contrast than equivalent light characters on a dark background (negative display). In effect, backlighting of the positive image “washes” around the dark characters to make them appear narrower. To counteract this phenomenon, the stroke width of the positive image should be approximately 20 percent heavier than that of the equivalent negative image.

Glare on the face of CRTs is one of the most frequently cited problems associated with VDT operation (Fig. 5). Glare from uncontrolled ambient light reflects into the operator’s eyes, making it difficult or impossible to distinguish images on the screen, as well as producing eye strain and fatigue. High ambient light also reduces the contrast between background and displayed characters, since it adds its energy to both. Antireflective coatings, tinted windows, louvered screens, and various other filter media can be used to minimize this problem. Some of these glare control solutions reduce ambient light bounce at the expense of contrast and image sharpness. The effectiveness of these

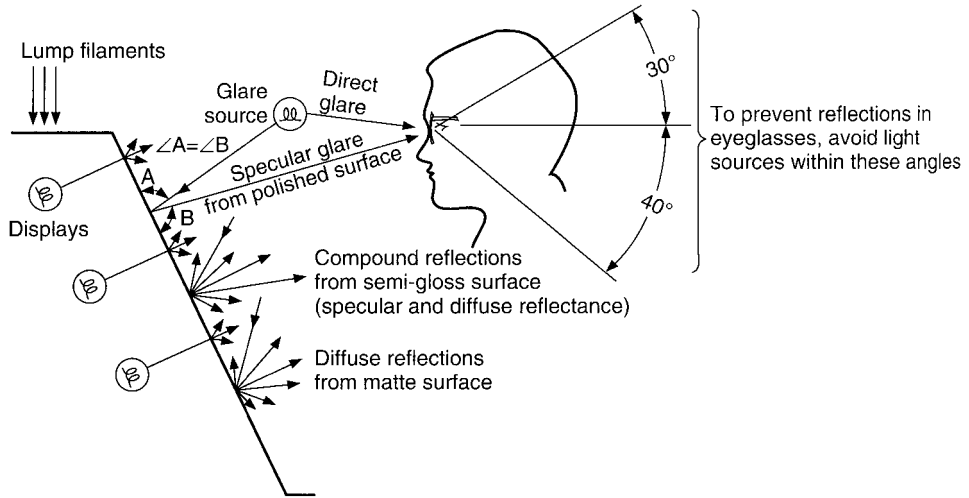


FIGURE 5 Glare from reflected ambient lighting can mask signals. (MICRO SWITCH.)

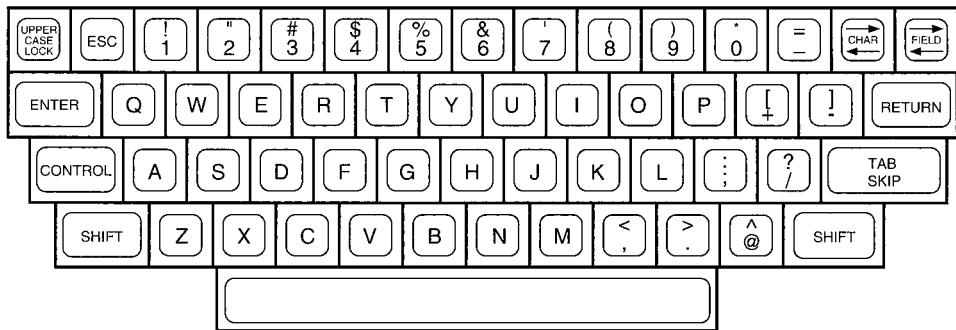


FIGURE 6 Keyboard with standard QWERTY layout.

various means should be evaluated carefully prior to making a commitment to use them. Further, the location and diffusion of ambient light sources should be investigated.

A wide range of image color is available on VDTs, but most use green, white, orange, or yellow contrasted with black. Any color image will satisfy the visual requirements of the display, so long as it does not fall at either end of the spectrum. Multicolor displays at these extreme wavelengths are also poorly perceived by the eye and should be avoided.

Keyboards

Current trends in programmable electronic data storage and retrieval systems have made the traditional typewriter keyboard all the more relevant as a critical interface between user and equipment (Fig. 6).

Key groupings for special functions and numeric entry have taken their place alongside the basic keyboard format. Full-function intelligent keyboards with integral microcomputer-based electronics may be encoded to meet user requirements.

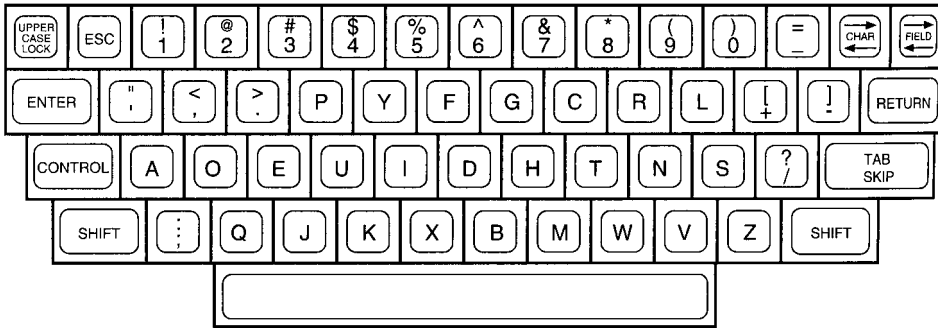


FIGURE 7 Dvorak simplified keyboard input.

The physical effort required should be consistent with the speed and efficiency of the operator and the duration of the operation. Without attention to these needs, the operator’s performance may suffer. Poorly conceived equipment standards have sometimes been established casually, or without meaningful studies of operator needs. With years of inertia, they become so ingrained in the user population that there is virtually no way of changing or even modifying their effect. The standard QWERTY keyboard (named by the arrangement of the first six keys on the upper row) is a prime example.

An alternative arrangement, the simplified keyboard (Fig. 7) was developed by Dvorak in 1936. It distributes the keys according to the comparative strength of the fingers and the frequency of letter occurrence in common English usage. This improved design has more than doubled typing speed and it is said to reduce the average typist’s daily keyboard finger travel from 12 to 1 mile (19.3 to 1.5 km). Adopting of the system has been and most likely will continue to be difficult because of the number of QWERTY machines in existence and by the extensive retraining of operators that would be required.

Volumes have been written about the optimum keyboard angle for efficient use. There have been keyboard layouts designed to operate at every angle between horizontal and vertical. However, there is a general consensus that somewhere between 10 and 20° is a comfortable workplane. Most contemporary keyboards fall within this angular reference. Recent studies have shown that by providing a keyboard with an adjustable setting for use between 10° and 25°, the needs of individual operators can be better accommodated.

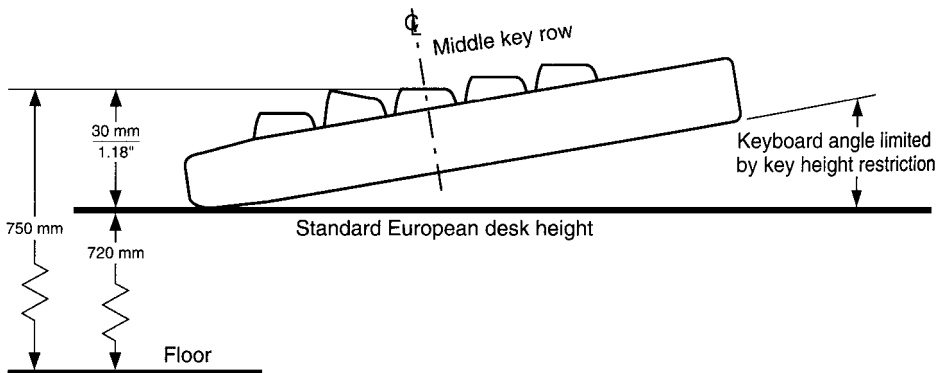


FIGURE 8 Basic DIN standard dimensions and angular restrictions. (MICRO SWITCH.)

The current European standard (Fig. 8) requires that the height of the keyboard measured at the middle key row should not exceed 30 mm (1.2 inches) above the counter top. Although there are a number of low-profile keyboards thin enough to allow meeting this requirement, the combination of the keyboard enclosure dimensions and the 30-mm DIN height standard may restrict the keyboard angle to 10° or less, and as stated previously, the preferred angle is between 10° and 20°.

DIN standards also recommend that if a keyboard is mounted at a height of more than 30 mm (1.2 inches), then a palm rest must be provided to reduce static muscle fatigue. This dictum, however, may be subject to question. A palm rest may be desirable for some low-speed data entry, but a high-speed typist does not rest palms on anything. The use of a palm rest could only promote inefficient and error-prone typing. In addition, a palm rest will require more desk space, which is usually at a premium.

For rapid keyboard entry, a 19.05-mm (0.75-inch) horizontal spacing between key centers is considered comfortable. For rapid entry it is desirable for traveling keys to have a displacement of between 1.3 and 6.4 mm (0.05 and 0.25 inch) along with an associated force of 25.5 and 150.3 grams (0.9 and 5.3 ounces). Traveling keys are preferred to low-travel touch panels, which are better suited for low-speed random entry situations. When used with a data display terminal, it helps to have the keyboard enclosure separable from the display, so the user can position it for maximum comfort.

Some of the more ambitious human factors efforts have created keyboards with a slight concavity, running from front to back on the surface of the combined key faces. This reflects the natural radii followed by the fingers moving up and down the key rows.

Other experimental keyboards have separated the left- and right-hand keys into two groups spaced slightly apart. The angle of each group parallels the natural angle of the hand and arm positioned in line and directly in front of a user. This minimizes the angular offset between hand and arm normally required for keyboard entry. These separated keyboards are not commercially available yet, but their invention indicates a continuing interest in improving the performance of keyboard operators.

Voice Recognition

Most voice recognition systems in current use are programmed to respond only to specific voices and accept a relatively limited command vocabulary. As speech processing technology continues to develop and as memory capabilities are expanded, these systems will be able to accept instructions from anyone. As the technology matures, voice recognition systems are expected to be able to recognize and identify both specific and nonspecific users. This process could eventually be used for authorized access to databases, and could supersede magnetic-card readers for clearance in security situations. More on voice recognition technology is given in a separate article in this handbook section.

ENVIRONMENTAL FACTORS

Numerous factors can affect the short- and long-term operating performance of the process or machine interface. Two of the most important of these are (1) the comparative gentleness or hostility of the local environment and (2) ambient lighting.

Local Environment

Industrial environments frequently pose a threat to the life and reliability of interface components. Often they are subjected to daily routines of abuse. Controls and displays may be splashed by water, oil, or solvents. They may be powdered with a layer of soot and dust, including metal particles, sticky vapors, and various granulated, gritty substances. However, even under these harsh circumstances, a resistant yet still attractive interface can be designed that will take advantage of oiltight manual controls, protective membrane touch panels, and ruggedized keyboards.

Ambient Light Conditions

External interface lighting almost always will either enhance or downgrade display visibility. As the ambient light level decreases, the visibility of self-illuminated displays will increase. For example, low-output lamps and projected color or “dead front” displays require low ambient light. Conversely, a brighter display is needed for recognition in high ambient light. By way of illustration, full indicator brilliance may be called for in the bubble of an aircraft cockpit exposed to direct sunlight, whereas a minimum glow from a display will be most appropriate at night. A situation like this will call for a brightness control for the indicator. Wherever possible, the designer should customize display brilliance to accommodate a wide range of brightness in any similar circumstances.

COGNITIVE SKILLS AND PROCESS CONTROL

Kai-Chun Cheng*

Ray E. Eberts*

INTRODUCTION

The process industry is a complex and dynamic system that involves the processing of multiple materials and energy to produce a new product. In addition, it often requires the integration of the efforts of many workers in the plant. Descriptions of conventional and modern process control system can be found in several references [1–4].

Because of the development of technology, automation has been widely incorporated into the design of process control systems. Process control is increasingly becoming a cognitive activity for the operator rather than a perceptual and control task. In a classification of the process operator's tasks by Lees [5], four of the five task categories are based on cognitive activities. Partly motivated by this, much research activity in human factors and cognitive psychology has been directed toward characterizing the cognitive skills developed by experts, how novices can be trained so that they develop those skills, and how equipment can be designed so that it fits in with the skills that operators have. An understanding of the cognitive skills possessed by operators can help in that design process. In addition, this understanding can also be used to train operators and to predict human performance in existing systems.

PROCESS CONTROL TASKS

The operator abilities needed for process control encompass the whole range of human abilities from perceptual motor tasks, such as tracking and manual control, to cognitive tasks, such as decision making and problem solving. Lees [5] provides an excellent review of the research on process control

* Purdue University, West Lafayette, IN 47907.

TABLE 8.1 Process Control Subtasks

Subtask	Description	References*
Monitor (vigilance, signal detection)	Monitor for seldom-occurring event; separate signal from noise	[7–9]
Control (tracking)	Keep system on optimal course	[10]
Interpret (categorization, quantization, estimation, filtering)	Separate random fluctuations of system from actual course; filter out noise	[11,12]
Plan (decisions, allocate resources, resource sharing)	Set goals and strategies; efficient use of resources; sequencing of tasks; heuristics; develop strategies	[13–15]
Diagnose	Identify the problem when a fault occurs	[16]

* Complete references can be found in Ref. 6.

operators. Several subtasks for process control in Table 8.1 have been extracted from the Lees article; this list of subtasks will be used throughout the article. The references for each of the subtasks describe characteristics of human performance and often suggest ways of improving performance. For example, the Peterson and Beach article [16] describes areas in which humans make poor decisions and therefore suggests ways to improve decision making by offering the right kind of information.

MODELS OF PROCESS CONTROL TASKS

The control technology had advanced from the conventional control techniques such as feedback and feedforward control to more advanced model-based predictive models and to recent developments such as adaptive control, artificial intelligence, and expert system [17, 18]. The goal was to develop models that can accurately capture the behavior of an actual system so that algorithms can be developed and used to design automatic process control systems.

Similarly, expert operators' behavior was studied and models were built to try to capture the characteristic of an expert. The goal was to learn how the experts could successfully control the processes and use this knowledge in the design of the control system. Thus novice operators can follow the expert operators' patterns and gain the expertise quickly. In addition, the experts' knowledge could help us design better training programs to help novice operators gain the desired expertise.

Quantitative Models

Model-based predictive controls are quantitative models that use dynamic mathematical formulas to describe the behavior of the system (references to several of the available models can be found in Ref. 6). In most cases, these models use physical system models and parameters to describe and understand human performance. Since many of the human tasks described in these models can now be performed automatically, the emphasis in modeling the operator has switched from these physical descriptions of performance to modeling the complex cognitive tasks required by current process control operators. These kinds of models are described in the following subsections.

GOMS and NGOMSL

GOMS (which stands for goals, operators, methods, and selection rules) and NGOMSL (natural GOMS language) models were formulated for human-computer interaction tasks. Since most of the

complex cognitive tasks required by process control operators are performed through interaction with a computerized process control system, the human–computer interaction model is appropriate. For GOMS [19], human information processing is described by four parameters: cycle time, the representation code, storage capacity, and decay time. Each task is then analyzed to determine how the operator processes the task information. The cycle time parameter can be used to make time predictions for completion of the task. The other parameters (representation code, storage capacity, and decay time) indicate the limitations of the human operator. When these limitations are compared with the task demands, error predictions can be made.

The NGOMSL model, developed by Kieras [20], is an extension of the GOMS model that can be used to predict the usability of a product or system. An operator’s interaction with a product or system is modeled by determination of the goals and the subgoals needed to accomplish tasks or subtasks, the methods required for accomplishing the goals, the steps (operators) to be executed in the methods, and the selection of rules when choosing between different methods. With the NGOMSL model, the steps that a person must go through to accomplish a task can be clearly listed, including both external and mental processes. Once the model is formulated, quantitative predictions can be made about the execution time, mental workload, learning time, and gains that are due to consistency. The operator–system interaction, when defined by the model, can be analyzed to find ways to improve performance of the operator, through system design changes, for the time needed to perform a task, the time needed to learn the system, the mental workload required for operating the system, and the consistency of procedures required of the operator.

GOMS and NGOMSL have been quite successful at accurately predicting human performance in computerized tasks. GOMS and NGOMSL can be applied to any process control tasks that require the operator to interact with a computer.

Expert Systems

Another approach to modeling the complex cognitive activity of process control operators is in terms of expert systems. At one level, expert systems can be differentiated from quantitative models in that the expert systems are heuristic and the quantitative models are algorithmic. An expert system is defined as an intelligent problem-solving computer program that uses knowledge and inference procedures to achieve a high level of performance in some specialized problem domain. This domain is considered to be difficult and requires specialized knowledge and skill. A common form of representing the knowledge in expert systems is to use production rules, which can be of the form

IF (premise) THEN (action)

The premise is a combination of predicates that, when evaluated by the program as true, lead to the specified action. The process control operator is modeled through the production rules used to control the process. KARL is an example [21] of an expert system developed for process control tasks.

Conclusions

The success of the applications of the above models has been mixed. The quantitative models are often based on models of physical systems and thus fail to capture abilities, such as flexible problem solving, that are uniquely human. These models are good at describing performance under ideal and optimal conditions (although not to the level found in most engineering models) and are also fairly good at performing control tasks; they fail if confronted with novel situations. The expert system models can deal with novel situations, although success of these models has been difficult to determine. GOMS and NGOMSL models have been shown to model human behavior accurately for those tasks in which the operator interacts with a computer.

EXTRACTING EXPERT KNOWLEDGE

Understanding expert process control operators is important in two ways. First, the purpose of a training program is to instruct a novice how to become more expert. Conveying how an expert performs tasks is important in training programs. Second, artificial intelligence techniques can be used, through computer programming, to make a process control system take on humanlike characteristics and intelligence. The knowledge or intelligence of experts must be determined in order to program such systems. In both cases, knowledge must be extracted from experts to use in training programs or artificial intelligence programs.

Unfortunately, extracting expert knowledge is often difficult. Experts generally store their knowledge and expertise implicitly in their mind rather than in a set of explicit rules to be followed. Thus expert operators generally have difficulty describing how they perform the control tasks. Sometimes, the expert's statement even contradicts his or her own behavior [22, 23]. The following techniques have been used for extracting expert knowledge.

Protocol Analysis

Following Newell and Simon's information processing theory [24], a technique for acquiring knowledge from experts was developed. This technique was referred to as the protocol analysis [25]. Protocol analysis requires an expert to solve a problem and at the same time verbalize the actions and/or thinking. The recorded protocols can then be analyzed to determine the thought process and information used by the expert. Another example of using protocol in extracting expert knowledge was an expertise transfer system [26] (ETS) developed based on Kelly's interview technique [27]. The ETS asks the expert series of "why" questions and uses the data to construct a knowledge base for the problem.

Scaling

Scaling techniques use statistical methods to calculate the spatial relationships between items of information and construct a hierarchical structure that represents an expert's knowledge base. Two methods are used often to scale the data, multidimensional scaling [28–30] (MDS) and cluster analysis [31, 32]. More detailed descriptions about these techniques can be found in the listed references.

Pattern Recognition

Expert knowledge can also be retrieved from the patterns in experts' behavior. These patterns could be how the experts interact with the system such as the when the display is checked, what type of information was checked, what action was taken, etc. When certain methods are used, the patterns of experts' behavior can be retrieved. Then, through the interpretation of these patterns, a model can be built that describes how experts control the system. Again, statistical tools could be used to retrieve the patterns in the recorded data. For example, Joyce and Gupta [33] and Gains, et al. [34] both had used the latencies between keystrokes to identify individual users. Another tool for identifying the patterns is by using neural networks. Past research in different areas has already shown that, when trained properly, neural networks have the ability to learn, memorize [35], and recognize patterns [36]. Kuespert and Mcavoy [22] had developed a framework for extracting expert knowledge through neural networks. A third tool for identifying the patterns in the data is to use filters. When a filter is used, patterns can be classified into different categories (e.g., Ref. 37).

EXPERTISE IN PROCESS CONTROL

Several characteristics can describe the expert process control operator. The first characteristic is that, after time, performing the task becomes automatic. Second, experts need accurate mental models of the system in order to control the system and make the right kinds of decisions. Third, operators must develop the ability to represent the system spatially.

Automatization

As a necessary, but not sufficient, condition, an expert can be characterized as someone who has been doing the task for a long time. Although it is difficult to determine when someone becomes an expert, one popular notion is that an expert has practiced the task for at least 5000 hours. The knowledge base accumulated by an expert is enormous. In a study of master chess players, Chase and Simon [38] estimated that these experts could recognize approximately 31,000 basic or primitive piece configurations. Brooks [39] has estimated that expert programmers have available to them between 10,000 and 100,000 rules that can be used to perform programming tasks. The expertise in a computerized expert system does not appear to be very expert in comparison. As a comparison, MYCIN, an expert medical diagnosis system and one of the most successful expert systems, contains 450 rules [40].

Researchers in expert systems claim that expert systems can, in essence, achieve the experience levels seen in true experts by acquiring expertise from several experts. The combined experience level would be greater than that of any one expert alone. Acquiring knowledge from an expert is difficult, as discussed above. Anecdotal evidence indicates that experts are often the least able to explain to novices their expertise. Lees [5] states that "it is well known, for example, that a pilot may be skilled in flying but may give a quite erroneous account of the control linkages in the aircraft." On a more empirical basis, Woodworth [41] cites evidence that conscious content of verbal reports disappeared with extended practice. Just at the time that, for example, a process control operator would be most useful to a knowledge engineer (one who gleans knowledge from an expert for an expert system), the expert operator may not be able to verbalize how to perform the task.

Accurate Mental Models

Experts are very good at knowing what to expect from the operation of the system. This ability is often referred to as the operator's having an accurate internal model of the system [42]. To have an accurate and effective internal model, the operator must be able to predict how the system will function. The events as they happen can then be compared with this prediction to see if anything unusual has occurred. Kragt and Landeweerd [43] characterized this ability as a "routine model" that the operators used when the process was performing in normal situations. Bainbridge [13] found that expert process control operators continually updated, in their heads, the current state of the system and where the system was moving. Without expectancies or without the internal model, the operator would have nothing on which to evaluate the current operation of the system.

Spatial Representations

Expert process control operators have available to them a spatial representation of the system. A plausible alternative to a spatial representation would be a propositionally based rule system similar to that used in expert systems. To differentiate between these two kinds of representations, consider a process in which the operator knows that the water level in a tank is dropping. If the operator stored a set of rules, he or she would mentally step through the rules to find one that fit; e.g., if the water level is dropping, an output valve is open. For a spatially based model, the operator would represent the task spatially as a physical system with locations and movements between the locations. To find the source of a problem, such as falling water level, the operator could mentally run a simulation of

the process until a match is found. As an example, the operator could picture the water flowing out of the tank through an open valve. When expert steam plant operators were interviewed by Hollan et al. [44] the experts indicated that they did mentally run such simulations to solve system problems. Rasmussen [16] found a similar phenomenon in expert process control operators and characterized this kind of activity in fault diagnosis and troubleshooting as a topographical search. Spatial representations of process control tasks, especially fault diagnosis, were also found by Landweerd [45] and by Wickens and Weingartner [46].

A spatial representation of a task is an efficient way to store information. By spatially representing a process control task, the operator must represent physical locations, must know how the systems can interact with each other, and then well-learned and versatile reasoning and problem-solving strategies can be used to make inferences about the process. As a common example of how such a representation scheme could work, if someone inquires about the number of windows in your house, this number is usually not stored explicitly. Rather, the number can be arrived at by mentally picturing yourself moving through the rooms of the house, counting as you go. Propositionally based rules can then be generated from this spatial information as a secondary process. Storing and retrieving 100,000 rules from memory, as an example, would be difficult; a spatial representation of the information is more efficient. A computer model of this kind of expertise, however, is difficult to generate. To generate one, we would have to know how to incorporate the spatial information in computer code, how to model the reasoning that occurs on this spatial information, and how particular strategies (such as picturing yourself walking through rooms) are chosen from the vast amount of strategies, many fruitless, that must be available. Some research is being done in this area to design expert systems that capture this kind of expertise [47].

TRAINING COGNITIVE SKILLS

Novices can acquire expert behaviors in two ways: by training, thus receiving the experience necessary, or by design of the system that assists novices to exhibit behaviors that are more expertlike.

Training is costly but cost effective in the long run, as shown in the example of the Inland-Orange Mill, which adopted a well-planned training program [48]. An operator properly trained and skilled at the task can perform many functions that machines are not capable of yet, such as decision making and flexibility in problem-solving techniques. It reduces downtime for the machinery and costly errors. Also, as Younkin and Johnson [49] emphasized, training is the key to a successful automated computer control system.

Many available techniques, which were not available a few years ago, can bring down the cost of training. These techniques fall into two areas: advances in cognitive skill training and advances in computerized training methods.

Using a Predictor

Several techniques can be utilized in training novices to gain the desired cognitive skills that expert operators possess. Providing a predictor of the future state of the system is one way to increase cognitive abilities. Recall that an integral part of the operator's internal model was the ability to predict where the system should be so that this prediction could be compared with the current state of the system. Predictors can be used in either the training process or as a job performance aid. After training, predictors on an augmented display were shown to increase the cognitive skills of operators [50]. The operators had learned to internalize the predictor so that performance would be high even without the predictor on the screen; this is advantageous if a breakdown occurs in the computerized aid. If augmentation is used during training, some trials should be made without the predictor so that the operator does not form a dependency on this assistance.

Spatial Information

Another method to develop cognitive skills is to provide the operators with spatial information during training. Recall that the expert operators appeared to conceptually represent a process control task spatially, and thus enhancing this ability through training would be useful. Steamer is an example [51] of how spatial information can be used when training operators. Steamer is a computer-assisted instruction program developed for the U.S. Navy to train their steam plant operators. A steam plant is used to power large ships and it typically has approximately 1000 valves, 100 pumps, and various turbines, switches, gages, dams, and indicators. The operational procedures for these kinds of plants are contained in several volumes. In an analysis of experts performing the steam plant operations, Hollan et al. [44] concluded that the experts had a spatial representation of the plant that was used to solve problems. When problems occur, operators could run mental simulations by using this internal spatial representation of the plant.

Analogies and Metaphors

A third method to develop cognitive skills is to use analogies and metaphors to train novices about how a system works. With analogies and metaphors, novices learn how knowledge of a similar situation can be applied to the new situation [52]. As an example, one of the most used and successful metaphors has been the desktop metaphor. Our knowledge of how to manipulate objects on our own desk (such as pen, paper, files, etc.) was applied to the design of the computer desktop. Another example can be found in the article by Mayer and Bromage [53]. They proved that training subjects by using concrete models could help them perform better on tasks that require transfer of their knowledge to novel situations.

Active Learning

A fourth method for developing cognitive skills is to encourage active learning. Young [54] and Kessel and Wickens [55] showed that operators could detect the change in system control functions better if they had active control of the system instead of passively monitoring the system. Developments in learning theory also shows the importance of active learning. Constructivism [56] states that learners need to build their own knowledge instead of purely receiving it from the instructors. Building of the knowledge happens when learners actively interact with their environment and restructure their previous knowledge. Thus effective training must encourage learners to interact with the system actively.

Learning Styles

Finally, as indicated by Dixon [57], one of the important considerations when designing a training program is to determine the individual learning styles of the learners. Some people may have high spatial abilities and others may have high verbal abilities. In addition to these innate abilities, other factors that lead to different learning styles include personality, cultural background, and learning speed. An effective training program should be able to adapt to each individual's needs and provide appropriate help and directions.

IMPACT OF AUTOMATION

People make errors. Automation has been very successful at reducing these errors although it may just be relocating human error to another level: Errors can still occur in setup in manufacturing and

maintenance and in programming. A person is still needed for performing cognitive-based tasks, as another system check, and to provide needed flexibility for unexpected events. A problem with automation is that it moves the human operator further away from the control loop. The issues associated with automating a process control task are considered further in the following subsections.

Vigilance Decrement

As more tasks become automated, the human operator is usually forced to monitor an automated system, and thus the monitoring load for the operator is increased. People are not very good monitors. This was first noticed when researchers studied radar operators during World War II [7] and found that the longer an operator continued at a task the more targets were missed; this became known as the vigilance decrement. With these long and monotonous tasks, the level of alertness decreases [58]. Solutions to this problem are to automate monitoring, provide special training, or provide effective feedback to the operators. Automating the monitoring is not a good solution because someone would then have to monitor the automatic monitoring. Special training is possible [59], and feedback, such as unexpected drills, should always be provided.

Out-of-Loop Familiarity

A second problem with automation in regard to cognitive skills is out-of-the-loop familiarity. As more inner-loop functions are automated, the operator is required to function more at the outer-loop stages. A problem occurs when an inner-loop function fails and the operator is required to jump in the loop and find the failure. Airlines have noticed that pilots have poor transfer from highly automated wide-body planes to smaller planes that require the inner-loop control [60]. Also, because much of the training of an operator is acquired on the job, automation can make learning more difficult. As mentioned above, Kessel and Wickens [55] and Young [54] found that operators who developed an internal model of the system from monitoring the system were not very good at detecting system faults. Operators who developed an internal model from controlling the system were much better at detecting system faults. This implies that skill maintenance of the operators may be required. Having the operators control the process in a simulator may be an important way to develop and maintain cognitive skills that may be at lost because of automation.

Overestimating the Intelligence of the Computers

A third problem with automation, especially with the incorporation of expert systems, is that the operator may think that the system is more intelligent than it actually is. Wickens [60] presents two examples of this occurring in aviation. In Detroit, Michigan, in 1975, a DC10 and an L-1011 were on a collision course; the air traffic controller saw this but did not act because he knew that if a collision became imminent the system would alert him. About this time, the air traffic controller took a break. The new controller saw the collision course and managed to contact the DC10 in time to just barely avert the crash. In another example, a crash of an L-1011 in 1972, the pilot put too much faith in the autopilot. When a fault occurred in the plane, the pilot put the plane on autopilot so he could diagnose the fault. He did not monitor the autopilot, the autopilot did not hold, and the plane crashed. Although many factors could contribute to the accidents, it appears that an important factor in both accidents was on overreliance on automation. The solution to these problems may not be less automation but could instead be to enable the operators to be aware of the limitations of the automation.

This problem also occurs in process control tasks. Jenkins [61] ran a simulation in which an operator along with an expert system controlled a nuclear power plant. He found that the operators assumed that the expert system knew more than it actually did; they thought that the system should have known when a failure in the cooling system occurred. In another computerized task, Rumelhart

and Norman [62] found that novices often attribute humanlike characteristics to the computer and cannot understand when the computer makes nonhumanlike mistakes. If incorporating expert systems into process control, the human operator should be thoroughly trained on what the system does and does not know.

Information Overload

As systems become automated and processes are integrated, the information to be perceived and processed by each operator increases. One solution to this problem is a software program that would be capable of transforming the large amount of data into useful, manageable, and easily perceived and understandable formats. Graphical representation is a powerful tool for displaying large amounts of data in a meaningful way. It could enable operators to detect changes in the system status, interpret the recorded measurements, and achieve better retention [63–65].

In addition, a fault detection and analysis system could be used to detect process faults. A fault detection and analysis system can help operators diagnose the possible faults in the processes and thus reduces the operator's workload. Early detection of the process faults could help prevent accidents, improve safety, and increase efficiency and productivity. Shirley [66] and Sassen et al. [67] have provided examples of such fault detection systems.

CONCLUSIONS

Existing models of process control operators were investigated. Quantitative models were useful for modeling the physical responses of operators. GOMS and NGOMSL models, which were formulated for modeling human–computer interaction tasks, are useful for modeling the cognitive activity of operators required to interact with computers in process control systems. Expert systems can be used to capture some of the knowledge required of operators for performing complex and unanticipated tasks.

Methods to extract knowledge from experts were also summarized. This knowledge can be used for training novices to become more skilled or for making the process control system more intelligent by using human intelligence to program expert systems. One conclusion from this analysis is that a human expert operator has a huge database of knowledge that is used to operate a process control plant. Only a small subset of this knowledge can ever be extracted from the operator. Completely automating a process control system, in which the system performs the intelligent functions now performed by the operator, is not yet possible. An operator is still needed, although the role of the operator is changing.

Training of operators in process control is important. With a computer, we can copy an original disk to a new disk so that the new disk has the same information as that of the original disk. We cannot, however, merely copy the expert knowledge from an expert to a novice through knowledge extraction techniques and training programs. We can, on the other hand, study experts to find characteristics of how they store and process information about process control. This information for experts appears to be organized in a spatially based mental model that can be run, similar to running a simulation on a computer. This mental model is developed through active interaction with the system or a simulation of the system. Training programs should concentrate on methods that help develop these accurate mental models.

Finally, process control tasks are becoming more automated. Although automation can remove human error from some aspects of the system, automation provides other opportunities for human errors to occur. These potential problems with automating systems were outlined. Understanding these problems will help in avoiding them through proper training programs or further refinements in the design of process control systems.

REFERENCES

1. Edwards, E., and F. Lees, *The Human Operator in Process Control*, Taylor & Francis, London, 1974.
2. Patrick, D., and S. Fardo, *Industrial Process Control Systems*, Delmar, Albany, New York, 1997.
3. Murrill, P. W., *Fundamentals of Process Control Theory*, 2nd ed., Instrument Society of America, Research Triangle Park, North Carolina, 1991.
4. Polke, M., ed., *Process Control Engineering*, VCH, New York, 1994.
5. Lees, F. P., "Research on the process operator," in *The Human Operator in Process Control*, E. Edwards and F. P. Lees, eds., Halsted, New York, 1974, pp. 386–425.
6. Eberts, R. E., S. Y., Nof, B. Zimolong, and G. Salvendy, "Dynamic process control: cognitive requirements and expert systems," in *Human-Computer Interaction*, G. Salvendy, ed., Elsevier, Amsterdam, 1984, pp. 215–228.
7. Mackworth, N. H., *Q. J. Exp. Psychol.* **1**, 5–61, 1948.
8. Swets, J. A., Ed., *Signal Detection and Recognition by Human Observers: Contemporary Readings*, Wiley, New York, 1964.
9. Mackie, R. R., *Vigilance: Relationships among Theories, Physiological Correlates, and Operational Performance*, Plenum, New York, 1977.
10. Poulton, E. C., *Tracking Skill and Manual Control*, Academic, New York, 1974.
11. Rouse, W. B., "A model of the human in a cognitive prediction task," *IEEE Trans. Syst. Man Cybern.* **SMC-3**, 473–477, 1973.
12. Rouse, W. B., "Human perception of the statistical properties of discrete time series: effects of interpolation methods," *IEEE Trans. Syst. Man, Cybern.* **SMC-6**, 466–472, 1976.
13. Bainbridge, L., "Analysis of verbal protocols from a process control task," in *The Human Operator in Process Control*, E. Edwards and F. P. Lees, eds., Halsted, New York, 1974, pp. 285–305.
14. Beishon, R. J., "An analysis and simulation of an operator's behavior in controlling continuous baking ovens," in *The Simulation of Human Behavior*, F. Bresson and M. deMontmollen, eds., Durod, Paris.
15. Peterson, C. R., and L. R. Beach, "Man as an intuitive statistician," *Psychol. Bull.* **68**, 29–46, 1967.
16. Rasmussen, J., "Models of mental strategies in process plant diagnosis," in *Human Detection and Diagnosis of System Failure*, J. Rasmussen and W. B. Rouse, eds., Plenum, New York, 1981, pp. 241–258.
17. Asbjonsen, O. A., "Control and operability of process plants," *Compu. Chem. Eng.* **13**, 351–364, 1984.
18. Fisher, D. G., "Process control: an overview and personal perspective," *Can. J. Chem. Eng.* **69**, 5–26, 1991.
19. Card, S. K., T. P. Moran, and A. L. Newell, *The Psychology of Human Computer Interaction*, Erlbaum, Hillsdale, New Jersey, 1983.
20. Kieras, D. E., "Towards a practical GOMS model methodology for user interface design," in *Handbook of Human-Computer Interaction*, M. Helander, ed., Elsevier, Amsterdam, 1988, pp. 67–85.
21. Knaeupper, A., and W. B. Rouse, "A model of human problem solving in dynamic environment," in *Proceedings of the 27th Annual Meeting of Human Factors Society*, Human Factors Society, Santa Monica, California, 1983, pp. 695–703.
22. Kuespert, D. R., and T. J. Mcavoy, "Knowledge extraction in chemical process control," *Chem. Eng. Commun.* **130**, 251–264, 1994.
23. Greenwell, M., *Knowledge Engineering for Expert System*, Ellis Horwood, Chichester, West Sussex, England.
24. Newell, A., and H. Simon, *Human Problem Solving*, Prentice-Hall, Englewood Cliffs, New Jersey, 1972.
25. Ericsson, K. A., and H. A. Simon, "Verbal reports as data," *Psycholog. Rev.* **87**, pp. 215–251, 1980.
26. Boose, J., "A framework for transferring human expertise," in *Human-Computer Interaction*, G. Salvendy, ed., Elsevier, Amsterdam, 1984, pp. 247–254.
27. Kelly, G. A., *The Psychology of Personal Constructs*, Norton, New York, 1955.
28. Kruskal, J. B., and M. Wish, "Multidimensional Scaling," Sage, Beverly Hills, California, 1978.
29. Shepard, A., K. Romney, and S. B. Nerlove, eds., *Multidimensional Scaling: Theory and Applications in the Behavioral Sciences*, Seminar Press, New York, 1972.
30. Cox, T. F., and M. Cox, *Multidimensional Scaling*, Chapman & Hall, London, 1994.
31. Tryon, R. C., and D. E. Bailey, "Cluster analysis," McGraw-Hill, New York, 1970.

32. Everitt, B., *Cluster Analysis*, Halsted, New York, 1993.
33. Joyce, R., G. Gupta, "Identify authentication based on keystroke latencies," *Commun. Assoc. Comput. Mach.* **33**, 68–176, 1990.
34. Gains, R., W. Lisowski, S. Press, and N. Shapiro, "Authentication by keystroke timing: some preliminary results," *Rand Rep. R2 56-NSF*, Rand Corporation, Santa Monica, California.
35. Chakravarthy, S. V., J. Ghosh, "Neural network-based associative memory for storing complex-valued patterns," *Proc. IEEE Int. Conf. Systems, Man Cybernetics* **3** 2213–2218 (1994).
36. Lo, S. B., J. S. Lin, M. T. Freedman, and S. K. Mun, "Application of artificial neural networks to medical image pattern recognition: detection of clustered microcalcifications on mammograms and lung cancer on chest radiographs," *J. VLSI Signal Process.* **18**, 263–274 (1998).
37. Siochi, A. C., and D. Hix, "A study of computer-supported user interface evaluation using maximal repeating pattern analysis," in *Human Factors in Computing Systems Conference Proceedings, CHI'91*, Association for Computing Machinery, New York, 1991, pp. 301–305.
38. Chase, W. G., and H. A. Simon, "Perception in chess," *Cognitive Psychol.* **4**, 55–81 (1973).
39. Brooks, R. E., "Towards a theory of the cognitive processes in computer programming," *Int. J. Man-Machine Stud.* **9**, 737–751, 1977.
40. Cendrowska, J., and M. A. Bramer, "A rational reconstruction of the MYCIN consultation system," *Int. J. Man-Machine Stud.* **20**, 229–317, 1984.
41. Woodworth, R. S., *Experimental Psychology*, Holt, New York, 1938.
42. Veldhuyzen, W., and H. G. Stassen, "The internal model—what does it mean in human control?" in *Monitoring Behavior and Supervisory Control*, T. B. Sheridan and G. Johannsen, eds., Plenum, New York, 1976, pp. 157–171.
43. Kragt, H., and J. A. Landeweerd, "Mental skills in process control," in *The Human Operator in Process Control*, E. Edwards and F. P. Lees, eds., Halsted, New York, 1974, pp. 135–145.
44. Hollan, J. D., A. Stevens, and N. Williams, "STEAMER: an advanced computer assisted instruction system for propulsion engineering," paper presented at the Summer Simulation Conference, Seattle, Washington, 1980.
45. Landeweerd, J. A., "Internal representation of a process fault diagnosis and fault correction," *J. Appl. Ergonom.* **22**, 1343–1351, 1979.
46. Wickens, C. D., and A. Weingartner, "Process control monitoring: the effects of spatial and verbal ability and current task demands," in *Trends in Ergonomics/Human Factors II*, R. E. Eberts and C. G. Eberts, eds., North-Holland, Amsterdam, 1985, pp. 25–32.
47. De Kleer, J., and J. S. Brown, "Mental models of physical mechanisms and their acquisition," in *Cognitive Skills and Their Acquisition*, J. R. Anderson ed., Erlbaum, Hillsdale, New Jersey, 1981, pp. 285–305.
48. Meadows, K., "The making of a training program," *Intech* **37**, 172–173, 1990.
49. Younkin, C. S., and F. B. Johnson, "Operator training: the key to successful computer control," *J. Water Pollut. Control Federation* **58**, 944–948, 1986.
50. Eberts, R. E., and W. Schneider, "Internalizing the system dynamics for a second order system," *Hum. Factors* **27**, 371–393, 1985.
51. Stevens, A., B. Roberts, L. Stead, "The use of sophisticated graphics interface in computer-assisted instruction," *IEEE Comput. Graphics Appl.* **3**(2), 25–31, 1983.
52. Gentner, D., "The mechanism of analogical learning," in *Similarity and Analogical Reasoning*, S. Vosniadou and A. Ortony, eds., Cambridge U. Press, New York.
53. Mayer, R. E., and B. K. Bromage, "Different recall protocols for technical texts due to advance organizers," *J. Education. Psychol.* **72**, 209–225, 1980.
54. Young, L. R., "On adaptive manual control," *IEEE Trans. Man-Mach. Syst.* **MMS-10**, 229–331, 1969.
55. Kessel, C. J., and C. D. Wickens, "The transfer of failure-detection skills between monitoring and controlling dynamic systems," *Hum. Factors* **24**, 49–60, 1982.
56. Blais, D. M., "Constructivism: a theoretical revolution in teaching," *J. Developmental Education* **11**, 2–7, 1988.
57. Dixon, N., "Incorporating learning style into training," *Training Development J.* **36**, 62–64, (1982).
58. Moore-Ede, M., S. Campbell, and T. Baker, "Effect of reduced operator alertness on the night shift on process control operator performance," *Adv. Instrumen. Proc.* **44**, 967–970, 1989.
59. Fisk, A. D., and W. Schneider, "Control and automatic processing during tasks requiring sustained attention: a new approach to vigilance," *Hum. Factors* **23**, 737–758, 1981.

60. Wickens, C. D., *Engineering Psychology and Human Performance*, Merrill, Columbus, Ohio, 1984.
61. Jenkins, J. P., "An application of an expert system to problem solving in process control displays," in *Human-Computer Interaction*, G. Salvendy, ed., Elsevier, Amsterdam, 1984, pp. 255-260.
62. Rumelhart, D. E., and D. A. Norman, "Analogical processes in learning," in *Cognitive Skills and Their Acquisition*, J. R. Anderson, ed., Erlbaum, Hillsdale, New Jersey, 1981.
63. Robertson, P. K., "A methodology for scientific data visualization: choosing representations based on a natural scene paradigm," in *Proceedings of Visualization '90*, IEEE Computer Society, Los Alamitos, California, 1990, pp. 114-123.
64. Schutz, H. G., "An evaluation of formats for graphic trend display—experiment II," *Hum. Factors* **3**, 99-107, 1961.
65. Standing, L., J. Conezio, and R. N. Harber, "Perception and memory for pictures: single trial learning of 2560 visual stimuli," *Psychonom. Sci.* **19**, 73-74, 1970.
66. Shirley, B. S., "An expert system for process control," *Tappi J.* **70**, 53-56, 1972.
67. Sassen, J. M., E. Buiel, and J. H. Hoegee, "A laboratory evaluation of a human operator support system," *Int. J. Hum. Comput. Stud.* **40**, 895-931, 1994.

DISTRIBUTED DISPLAY ARCHITECTURE

Since the early 1970s instrument displays have been undergoing numerous changes that reflect and incorporate the marked progress that has occurred throughout instrumentation and control technology. These would include development of the early minicomputers and the later introduction of the personal computer (PC); the advent of the microprocessor; advances in cathode-ray tube (CRT) technology and other display methodologies, such as liquid-crystal displays (LCDs), vacuum fluorescent displays (VFDs), and plasma (gas discharge); the decentralization of control centers by way of distributed control; the development of advanced networking concepts; a better understanding of human factors as they relate to the operation of machines and processes; and the engineering of interactive graphics among numerous other advancements that have seriously affected display practices, the general result of which is a severely altered interface. The contrast is dramatically apparent by comparing Figs. 1 and 2.

Contemporary systems are quite software-intensive. Installation and maintenance require different skills than in the case of prior-generation systems. Fewer electricians and hardware technicians are needed, whereas more expertise is required in the system engineering and software areas.

THE GRAPHICS INTERFACE

"Pictures" help immeasurably to improve an operator's comprehension of what is going on in a controlled process or machine. This, of course, is wisdom known since antiquity. It was first applied in industrial instrumentation in the 1940s, when elaborate process or manufacturing diagrams were put on permanent panels, with illuminated indicators placed directly on the diagrams, and where attempts were made to locate single-loop controllers to reflect the relative geometry of where the measurement sensors were located in the process. Such panels became plant "showpieces" and often

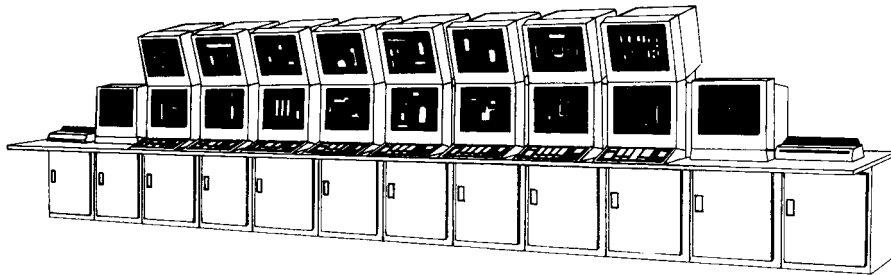


FIGURE 1 Long panelboard containing indicators and controllers used in large processing plant. Generally this type of panel was used in the early 1970s. The very beginnings of the cathode-ray tube (CRT) displays for annunciators are indicated by the CRTs in the upper right-hand portion of the view.

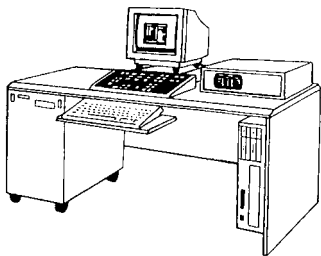
were constructed of costly enameled steel. Unfortunately changes were inevitably made in the process or instrumentation, and the beautiful initial panel soon was marked over with tape. A cluttered appearance progressively emerged, which hardly could be considered an inspiration to the operator's confidence in the system. Use of a CRT as the panel face essentially solved this problem of panel obsolescence. Later embellished through the use of color, it offered immense flexibility, not only at one location, but at several locations along a network. Currently these properties are being exploited further through the use of the X-window and X-network concepts. But the CRT has had some limitations too, notably the comparatively small space in which an entire control panel must be projected. Even though through menu techniques a process may be shown in detailed scenes, some loss of comprehension occurs, brought about by crowding and by the complexities introduced when expanding a scene into subscenes ad infinitum. Larger basic presentation techniques (such as flat screens) are constantly being refined.

Graphic displays are formed by placing a variety of symbolic forms, originally created by an artist and later reproduced electronically, on the display surface where they can be viewed by the user. These may enter the system by way of certain display standard protocols, such as GKS/PHIGS (graphical kernel system/programmer's hierarchical interactive graphics system), or otherwise software created. In some systems the objects are formed by using the basic output primitives, such as text, lines, markers, filled areas, and cell arrays.

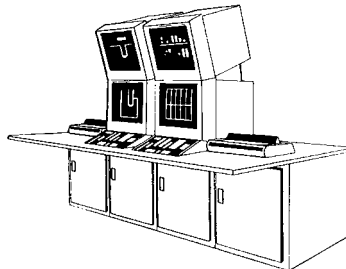
The display software must provide additional higher-level graphic objects formed with the basic primitives. For graphic displays on process systems, the fundamental types of objects available should



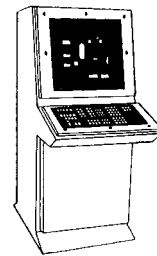
(a)



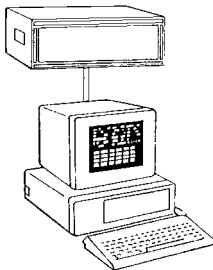
(b)



(c)



(d)



(e)



(f)



(g)

FIGURE 2 Examples of how the flexibility of CRT-based panels contributes to various interface configurations and distributed display architecture. (a) Central configurable CRT station with distributed historian. (b) Engineer's workstation. (c) Area CRT station. (d) Field-hardened console. (e) Independent computer interface (serial-interface-personal-computer). (f) Batch management station. (g) General-purpose computer.

include background text, bar graphs, trends, pipes, and symbols. It is desirable to allow each of the fundamental types to be freely scaled and positioned. It is also desirable for the trends to be displayable as either point plots or histograms, and for both text and trend fields to be scrollable together if there are more data available than can be reasonably displayed in the space allocated for the object. The system should provide libraries of symbols and facility for the user to edit.

Well-designed graphics displays can show large amounts of dynamic information against a static background without overwhelming the user. The objects used to form the static portion of the display

may be as simple as textual headings for a table of dynamic textual data, or a graphic representation of a process flow diagram for a section of a process, with dynamic information positioned corresponding to the location of field sensors. In a GKS system the static portion of a display usually is stored in a device-independent form in a metafile; in PHIGS, an archive file may be used. From there it can be readily displayed on any graphics device in the system, including various terminals and printers.

Display software also can provide the ability to define “pickable” objects and a specific action to take if an object is picked. Picking an object may call up a different display, show some additional information on the same display, or initiate an interaction sequence with the user.

The graphic display software should provide the ability to update dynamic data objects at specifiable time intervals, upon significant changes, or on demand. The update specification should be on a per object rather than a per display basis. The updating of each object should involve the execution of arithmetic expressions with functions available to access the database of the process information system. Besides changing the values of the data to be represented, the updating could result in changes to the attributes of the primitives used to form the objects.

For some users, especially process operators, color can add an important dimension to graphic displays, expanding the amount of information communicated. Color graphics are used widely today. (One must make certain that there are no color-blind operators.) Making hard copies of color displays has improved quite a bit during recent years.

USER INTERACTIONS

The most important aspect of any interface is the means by which the operator can interact with the system. This is the mechanism that the user may employ when accessing, manipulating, and entering information in the system. It is the key point in any interface design and in specifying requirements prior to procurement.

It has been established that best acceptance occurs when an operator becomes confident that there is no harm in using an interactive graphics system. To achieve this, software should be provided and used that makes certain that all interactions are syntactically consistent and that all data entered will be checked for validity. The system should reject invalid inputs and provide a message that the user can readily understand without external reference material.

The user interface must be responsive. The goal should be that all system responses to user actions occur faster than it is physically possible for the operator to perform a subsequent action. Some situations require more responsive interfaces than others. In general, the number of interactions and the complexity of each interaction should be minimized, even though this may require additional sophisticated hardware. Most interactions with a process computer system can be performed in an easier fashion with an interactive graphics system than with a textual dialogue system. However, even with interactive graphics, some textual interaction sometimes may be necessary.

“CONVERSATIONAL” INTERACTIONS

Historically systems have interacted by having the user carry out a textual dialogue in a dedicated portion of the display screen, sometimes called a dialogue area. The flow of the dialogue is similar to a vocal conversation—with the system asking for information with a prompt that the user reads in the dialogue area, and the user providing answers by entering alphanumeric text strings or pressing dedicated function keys on a keyboard. The system provides feedback to each user keystroke by echoing the user’s input to an appropriate position in the dialogue area, importantly of a different color than the system’s prompt. This is a natural form of interaction for the operator. The tools used to carry out such interactions are easy to construct.

The person who has programmed the system and the user should fully understand the dialogue language that is being exchanged. This can be troublesome because natural language communication

can be ambiguous, considering the multiple meanings of some words that arise from the level of education and sometimes the geographic background of the user. Programmers sometimes have difficulty in creating meaningful prompts that are short and easily accommodated in the dialogue area. Live voice systems also can be effective, but have not been accepted on a wide scale.

In process control systems the dialogue area frequently appears at the bottom of the display surface, allowing the user to view the displays while the interaction is progressing. In this way the user will not, in error, attempt to use old data left on the screen during an extended or interrupted interaction. If display updating is halted during interactions, the system should either obscure the data by performing the interaction in a dialogue box in the center of the screen, or in some way make it obvious that the displayed data are not reliable.

Dialogues should be designed to minimize the number of questions required and the length of user inputs needed. Prompts should be simple and easy to understand. They should list menus of possible alternatives when appropriate. Whenever possible, default values should be provided to facilitate the rapid execution of functions.

DATA INPUTS

An interactive graphic user's interface may consist of a bit-map graphic display device (usually with a dedicated processor), a keyboard, and one or more graphics input devices. Hence the electronic interface is essentially the same as the traditional interface, except that the video display unit has some additional capabilities and a graphics input device has been added. Normally, interactive graphics capabilities will be in addition to those of the conventional panel.

Graphics input devices are a combination of hardware and software that allows the positioning of a graphic pointer or cursor to a desired location on the screen and "triggering" an event to perform some action based on the selected pointer position. The event normally is triggered by pressing or releasing a button. Examples of physical input devices that have been used include the light pen, joystick, track ball, mouse, graphics tablet, touchpad, touchscreen, and keyboards. Once a device is selected, one should concentrate on how it can be used effectively. Among the most commonly used are the keyboard, a mouse, or a touchscreen.

Keyboard. A keyboard should be available with an alphanumeric section for entering text and an array of function buttons for certain specialized functions. In contrast to nongraphic interactions, the alphanumeric section will be used infrequently. On process systems, the function button section of the keyboard is important even in a graphic environment, because buttons can provide immediate generation of commands or random access to important information. Sometimes there is a need for keyboards with in excess of 100 function buttons on a single workstation. Also, there may be a need for lamps associated with keys that can be used to prompt the user. Keys should provide some kind of feedback when actuated. Tactile feedback is desirable, but not necessary. Many operators are prone to favor tactile feedback. Visual feedback from every keystroke is essential.

Touchscreen. This is probably the simplest of the graphic input devices to use because it requires no associated button for triggering events. This can be done by sensing a change in the presence of a finger or some other object. A touchscreen occupies no additional desk or console space and appears to the operator as an integral part of the screen. The touchscreen is useful to naive operators for selecting one of several widely spaced items that are displayed on the screen. However, because of calibration problems, the curvature of the CRT screen, and its coarse resolution (that is, between screen and human finger), implementations using touchscreens must be considered carefully in advance—probably involving prior experience to get the "feel" of the system. From a human factors standpoint, it is important that the system be designed to trigger the event upon the user's removal of a finger from the screen rather than upon the initial contact of the finger. This allows the user to correct the initial (frequently inaccurate) position of the cursor by moving the finger until the graphic pointer is properly

positioned over the designated object on the screen, at which point the finger can be removed, thus triggering the selection of the object.

Touchscreens operate on several principles, but most require an overlay on the monitor. Touchscreen technologies include the following.

Resistive. Two conductive sheets separated by an array of clear, tiny elastic dots. The first substrate layer may be glass or plastic. This has a conductive coating (such as indium or tin oxide) that possesses uniform and long-term stability. The second layer (cover sheet) may be of polycarbonate or oriented polyester (Mylar) plastic, which has a conductive inner coating on the surface facing the substrate conductive coating. A voltage measurement indicates where the circuit is completed. Resistive screens indicate only a single point to the computer at the center of the area of contact. In discrete resistive touchscreens, one of the layers includes equally spaced horizontal rows, which are etched into the surface; the others are in the form of vertical lines (columns). Thus the two sheets form a grid. Discrete screens usually are application-specific. Analog versions allow total software control for the touch zones. Characteristics of resistive touchscreens are 4000×4000 resolution, low cost, very rapid response, and no limitation on stylus used.

Capacitive. These operate on the principle that high-frequency alternating current couples better between conductors than direct current. The screen system is controlled by a radio-frequency source (approximately 10 MHz), the outputs of which go to each corner of the screen. When the screen is touched, current passes from the screen through the operator's finger. The amount of current emanating from each corner of the screen relates to the location of the touch. Characteristics of capacitive touchscreens are good resolution (1000×1000 points per screen side), fairly high cost, very rapid response, and some limitations on stylus used. The screen must be shielded from internal electronics noise and external electromagnetic interference. Very good optical properties.

Surface Acoustic Wave. This is a relatively new technology. When surface acoustic waves (SAWs) propagate across a rigid material, such as glass, they are efficient conductors of acoustic (sound) energy—at precise speeds and in straight lines. Special transducer-generated SAW signals travel along the outside edges of the screen, where they encounter reflective arrays on the screen. Each array reflects a small portion of the wave. Through complex logic circuitry, the SAWs determine the coordinates of any pliable device (including a finger) that may press the screen surface to form a transient detent. Some drawbacks solved by this technology include a combination of transparency, ruggedness, and ease of installation. Characteristics of SAW screens are a modest resolution (15–100 points per inch), high cost, and very rapid response.

Piezoelectric. Pressure-sensitive electronics detect and determine touch location. A touch on the screen causes the screen glass to flex slightly, applying pressure to each of the stress transducers. This pressure is inversely proportional to the distance from the touch point. Each transducer generates a signal proportional to the amount of pressure applied. Signals are processed into *X* and *Y* coordinates for transmission. Sometimes vibrations in the environment may adversely trigger one or more transducers. Resolution is limited (80 to 90 touch points); response is very rapid.

Infrared. The principle used is interruption of a beam of infrared (IR) light. The system consists of a series of IR light-emitting diodes (LEDs) and phototransistors arranged in an optomatrix frame. The frame is attached to the front of the display. A special bezel is used to block out ambient light. An IR controller scans the matrix, turning on LEDs and the corresponding phototransistors. This is done about 25 times per second. Characteristics of IR systems include reasonable cost, any type of stylus may be used, and response is very rapid.

Mouse and Track-Ball Input Devices. These devices are used to drive a cursor on the graphics screen. To move a cursor, the mouse is moved across a hard surface, with its direction and speed of movement proportionally related to the cursor's movement on the display screen. A track ball operates on the same principle, but the movement of a ball mounted on bearings within the track ball positions the cursor. It can be thought of as an upside-down mouse. Both the mouse and the track ball have one or more buttons that are used for control. One button is used to indicate when the cursor has reached the proper location. One button is used to activate or deactivate the device. The screen cursor can be used to make menu selections from the display or to place graphic elements.

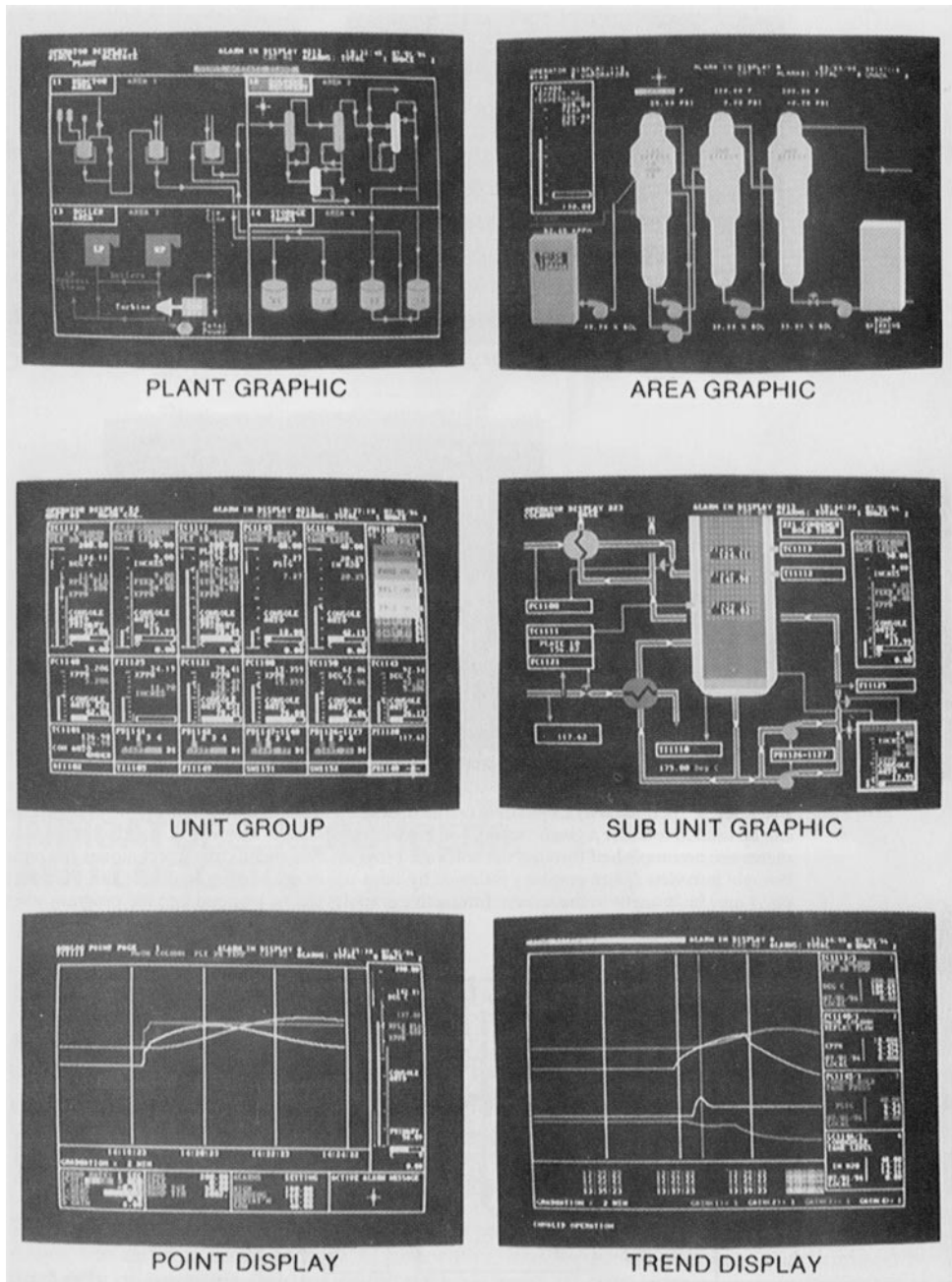


FIGURE 3 Depending on the operator's needs and preferences, the available area on a CRT screen can be used in total for a given "scene," or, more commonly, the area will be divided into windows. These arrangements are accomplished through the software program. Via menu-driven techniques the entire operation can be brought into view (plant graphic), followed by close-ups of areas, units, and subunits. Further, point and trend displays may be brought to the screen. Interactive graphics can be planned into the program wherever desired.

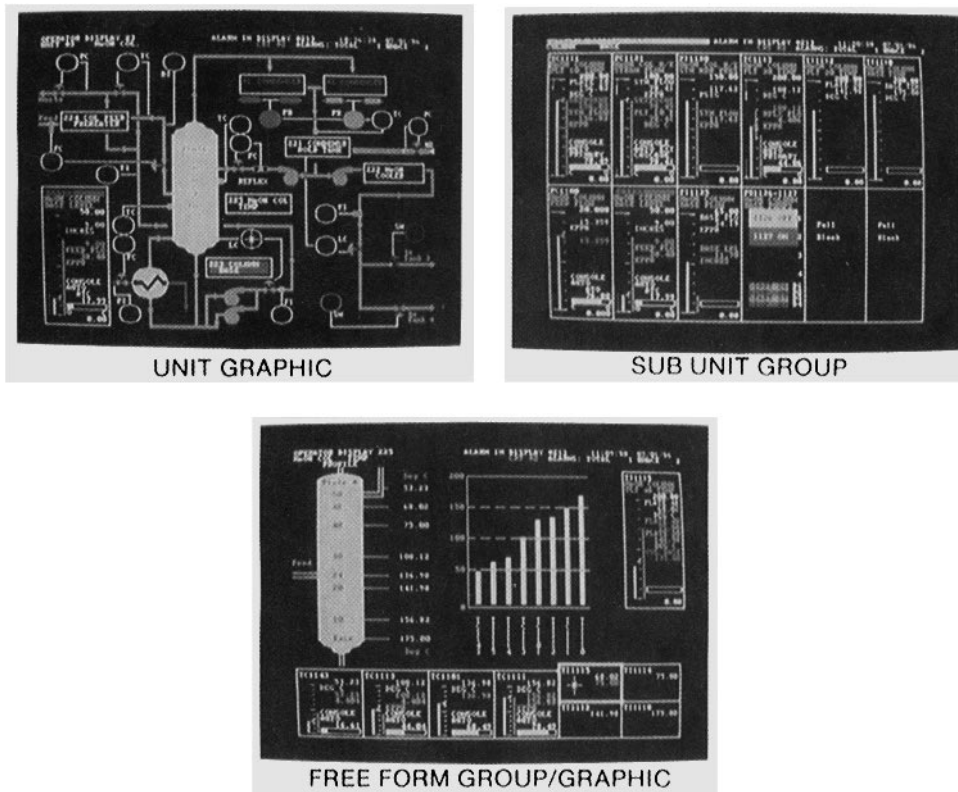


FIGURE 3 (continued) Depending on the operator's needs and preferences, the available area on a CRT screen can be used in total for a given "scene," or, more commonly, the area will be divided into windows. These arrangements are accomplished through the software program. Via menu-driven techniques the entire operation can be brought into view (plant graphic), followed by close-ups of areas, units, and subunits. Further, point and trend displays may be brought to the screen. Interactive graphics can be planned into the program wherever desired.

Digitizer. This has been a common input device, especially for computer-aided design (CAD) systems. Three main components of a digitizing tablet are pad, tablet, or surface; the positioning device that moves over the surface; and electronic circuitry for converting points into x - y coordinates and transmitting them to the computer. The device can be used to place a graphic element on the screen, make a menu selection from the screen, or make a menu or symbol selection from the digitizing tablet itself.

Joystick. Identical in function to the joystick supplied with home computer games, these devices are sometimes preferred when working with three-dimensional drawings because of their free range of motion. The joystick works on the same general principles as the mouse and track ball.

Light Pen. One of the very early means for interacting a narrow beam of light with photoreceptors behind the screen, it is similar in principle to some of the other systems described.

VISUAL DISPLAY DEVICES

Cathode-Ray Tube. The operating principles of this ubiquitous display device were well developed and the device was thoroughly seasoned by the time of its first use in industrial instrumentation and

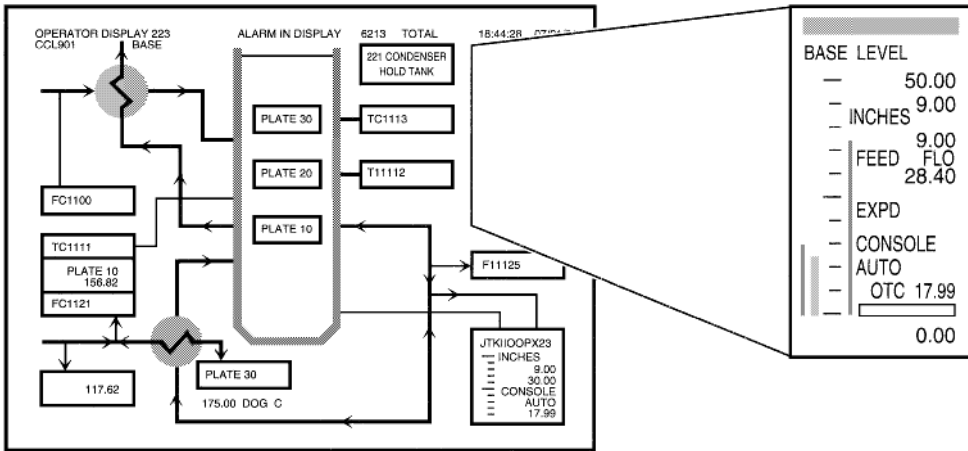


FIGURE 4 By way of software, a portion of a CRT display, namely, a window, can be singled out for detailed viewing. In the X-Window scheme, one or more windows could be designated as “dedicated” for use on an X-Network, making it possible for several stations along the network to obtain instant information that may be of interest to several operators in a manufacturing or processing network.

control systems. Of course, for many years it has been used in the form of the oscilloscope, where it enjoys a leading role in test instrumentation. In terms of process control and factory applications, it made its first appearance in CATV (closed-circuit television) applications, where it was used as a “seeing eye” for monitoring remote operations. Then it was later used in some of the earliest attempts to achieve several basic advantages of computer-integrated manufacturing.

The immense flexibility of the CRT is demonstrated by Figs. 3 and 4. Also as evident from other fields of CRT usage, such as animation and CAD applications, the CRT seems almost unlimited in its potential when in the hands of ingenious programmers and software engineers. See the articles, “Distributed Control Systems” in Section 3 and “Industrial Control Networks” in Section 7 of this handbook.

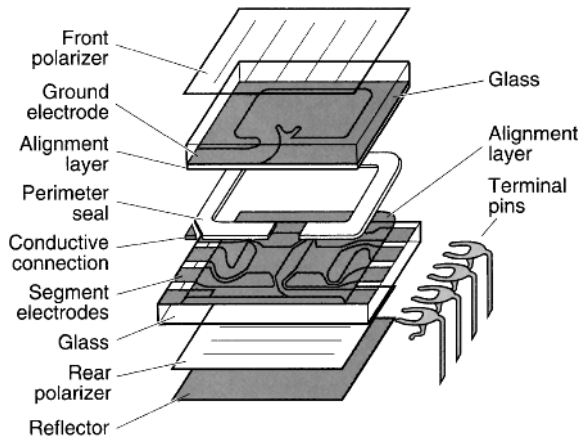


FIGURE 5 Cross-sectional view of typical TN (twisted nematic) liquid-crystal display with transparent indium—tin oxide electrodes patterned on glass substrates using photolithography techniques. (Standish Industries, Inc.)

Liquid-Crystal Displays. Introduced several years ago, liquid-crystal displays (LCDs) are achieving a measurable degree of design maturity, particularly after the addition of numerous colors, as indicated by the many thousands of units that are installed and ordered each year. LCDs are nonemissive and therefore require good ambient light at the proper angle for good viewing. The physical chemistry of LCDs is a bit too complex for inclusion here. However, a cross-sectional view of a typical LCD unit is shown in Fig. 5.

Electroluminescent Panels. These displays use a thin film of solids that emit light with an applied electric field. Because they are monochromatic, small, and expensive, they usually are applied when applications require a highly readable, lightweight, low-power-consuming visual display unit with very rapid update times. Design research in this field is continuing.

Vacuum Fluorescent Panels. These display units create thin images by accelerating electrons in a glass envelope to strike a phosphor-coated surface. They are bright, available in multiple colors, and have fast update times. Usually they are small in size and, therefore, find applications that require low-information displays, such as reprogrammable control panels or video-control stations, where graphics are simple and space is a premium consideration.

KNOWLEDGE-BASED OPERATOR TRAINING

G. K. McMillan*

Operator training systems are the key to successful implementation of new systems and the safe and efficient operation of existing control systems. Often the training system also serves the essential purpose of testing the control system design and configuration. The need for these operator training systems has dramatically increased because of the following:

1. Early retirement of experienced operators
2. Reduction in the number of operators
3. Ever-changing proprietary computer interfaces
4. Increased information displayed
5. Barrages of spurious alarms
6. Increased economic and environmental pressures
7. Aged and stressed process equipment
8. The need to push processes to their constraints
9. Increased product grades and decreased inventory
10. Increased sophistication and complexity of control systems
11. Loss of operator involvement from advanced automation

* Senior Fellow, Solutia Inc., St. Louis, Missouri.

For new plants, there is obviously a need to develop operator expertise in the process. What is less apparent is the great need to do the same in existing plants because of the loss of the experience base from years of corporate downsizing. This has affected both the operators and the engineers who provide technical and production support. Also, many projects to replace panel board instruments with distributed control systems (DCSs) or programmable logic controllers (PLCs) were justified based on head-count reduction. In the meantime, there was a constant evolution of operator interfaces for the DCS and the PLC. The differences between manufactures and even within a given manufacturer from generation to generation are significant. Finally, there is a new movement toward a Windows environment. While the Instrument Society of America standard for Fieldbus promises to create a common control language, there is no similar effort to establish a common interface.

The DCS has also increased the number of alarms by an order of magnitude over panel board instruments since a DCS typically comes with three free alarms whereas a panel board alarm required an annunciator window to be engineered, purchased, and installed. Unfortunately, most of these new DCS alarms are spurious. They pose a serious distraction and make the operator insensitive to real alarms. This, coupled with the dramatic increase in information from the additional process variables and diagnostics from smart transmitters, can cause information overload.

At the same time there is a tremendous increase in economic pressures from worldwide competition and the need to constantly increase profits and the return on equity to meet stockholder expectations and an increase in environmental regulations and voluntary initiatives to reduce emissions. Spills or inefficient operation that results in a loss in yield and increase in waste treatment are intolerable and make the difference between a plant's running or being sold or leveled. The need for restraint in capital spending means that many plants cannot replace existing equipment even though production targets have increased. Plants are typically running from 20% to 200% more than name-plate capacities with equipment that is 20 or more years old. Operating conditions are pushed closer to constraints (equipment or process limits on operating temperatures, pressures, levels, and flows) so that production rate can be increased without a loss in yield, safety, or on-stream time. The need to increase the product spectrum and coincidentally reduce product inventory leads to an increase in product changeovers that places additional stress on equipment and operations to maintain quality, yield, and capacity.

This combination of fewer and less-experienced engineers and operators with older equipment and higher production requirements is a recipe for a performance that does not meet expectations, which shows up in a loss of on-stream time, production rate, and yield. It is particularly noticeable in the time it takes to resolve a customer complaint, solve a production problem, or restart after a shutdown.

Advanced control systems that ride but do not violate constraints and provide automatic startups, changeover, and shutdowns can force a plant to meet production targets. However, this is a double-edged sword because the operator becomes too dependent on the control system. The more successful the implementation, the fewer the chances for the operator to develop and exercise his or her skills. The result is operator disengagement and eventually an inability for the operator to take over and run the process when the control system can no longer do its job because of changes, deterioration, or failures in the process, equipment, control valves, measurements, or controllers [1].

The following incremental improvements to operator training systems to help meet this increasing need are listed in order of increasing complexity.

1. Tiebacks
2. Ramps, filters, gains, dead times, and dead bands.
3. Liquid-material balances
4. Energy balances
5. Phase equilibrium
6. Vapor-material balances
7. Component balances
8. Charge balances
9. Reaction kinetics
10. Mass transfer

11. Polymerization kinetics
12. Electrolyte effects

The operator training system should have the following features. It should

1. Use the actual operator interface
2. Require no modification of the control system configuration
3. Test the complete plant
4. Test the complete control system
5. Ensure enough fidelity to show interactions and actual operating conditions
6. Ensure enough fidelity to provide process and control system knowledge
7. Show true process conditions for startups, changeovers, failures, and shutdowns
8. Automatically be updated for changes in process and equipment and the control systems
9. Provide faster than real time test runs of both the plant and the control system
10. Offer a freeze and bumpless restart of the plant model
11. Be capable of being quickly developed, tested, and demonstrated on a laptop by most process and process control engineers as opposed to modeling specialists
12. Become the source and depository of knowledge on the process and the control system

The simplest form of model for training is the tieback, in which the DCS or the PLC inputs are tied to the outputs by a simple relationship. For discrete control devices such as pumps, agitators, fans, and on-off valves, this tieback provides the motor run contacts and valve limit switches feedback inputs in the expected pattern based on the discrete outputs after a specified transition time. For control loops, the feedback measurement is the loop output multiplied by a gain. In some models, filters, dead times, and dead bands are added to provide more realistic dynamics. These parameters can be adjusted so that the loop responds in a fashion that is reasonable to the operator and can use loop tuning settings that are in the ball park. However, dynamics is not typically the forte of operators. While you might be able to satisfy the operators by an iterative correction of the response, there is very little assurance that the resulting dynamics is realistic. Tieback models reinforce these preset notions on the process and control system behavior. The main purpose of these models is to familiarize the operator with the DCS or the PLC interface. They also are simple enough that the whole plant can be simulated, and if the connections are between the model and the controller inputs and outputs, the whole configuration can be tested. While ideally the logic for the tieback model should be done independently of that of the configuration, the time savings in automatically generating these tiebacks from the configuration generally outweigh the loss of separation of the test and control system software. For critical loops and safety interlocks, the test software should be constructed independently by a different person from the one doing the configuration of the control or safety interlock system. Thus automatically generated tiebacks should be relegated to discrete device control and simple loops for which there are no safety, performance, or operational issues.

The use of ramp rates tuned to simulate levels is absurd in that they require more effort than a simple liquid-material balance and rarely provide a good representation of actual level response. Thus liquid and solid levels should be based on inventory or accumulation, which is the integral of the difference in flow into and out of a volume (mass balance). Gas pressure can be computed by the ideal gas law from the density of the vapor by extension of the mass balance to the vapor space. It may involve vaporization and condensation rates based on heat transfer and boiling points based on compositions. This necessitates the addition of energy and component balances. More of an issue than the additional complexity of the process model is the fast dynamics that makes it necessary to use special stiff integration methods, small step sizes, and/or larger than actual volumes [2].

The simulation of pH can be done from a charge balance based on dissociation constants and concentrations of acids and bases. For solutions that are not dilute, the changes in activity coefficients based on electrolyte effects should be included. Interval halving is used to find the pH where the

summation of positive and negative charges is nearly zero. This search routine converges rapidly. Alternatively, a polynomial fit to a titration curve can approximate pH from the ratio of reagent added to the vessel or stream. However, this does not take into account the shift of the titration curve from the change of the dissociation constants with temperature or the change in composition of the influent.

Reactors require kinetic expressions to determine the rate of component consumption and formation. Often an activation energy and preexponential factor are sufficient to show the dependence of the reaction rate on concentration and temperature. For vapor-phase reactions, pressure is also important. For polymerization, viscosity effects and chain initiation, building, and termination must be included. For copolymers and tripolymers, cross linking requires complexity beyond the capability of most models.

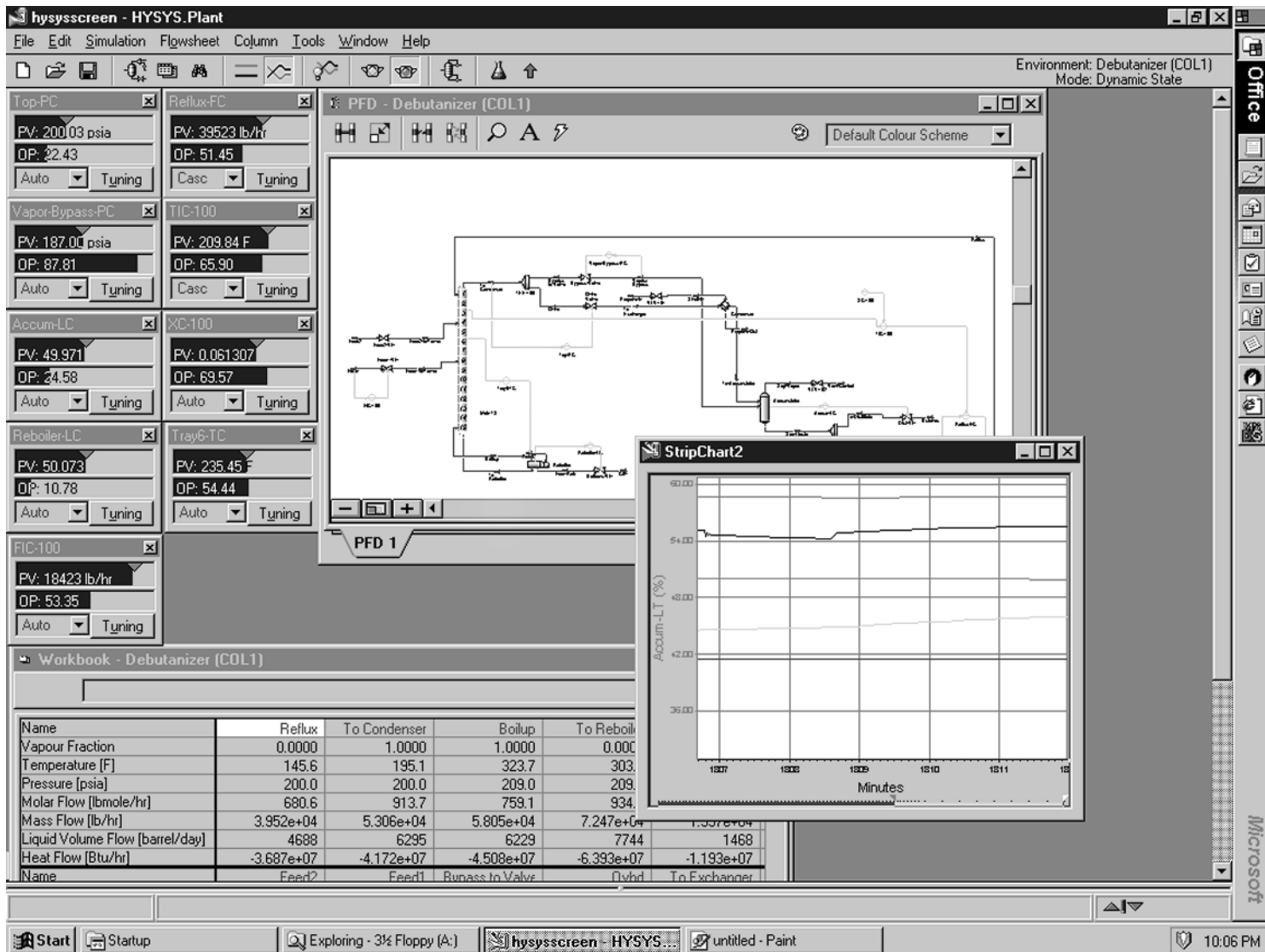
The additional fidelity from the inclusion of energy and component balances, phase equilibrium, and kinetics is necessary to provide the proper process gains, dynamics, and interactions necessary to convey process knowledge and test the control system performance. It extends the utility of the training system from just training operators to training engineers and technicians for operations, maintenance, and projects. It also opens the door to using the model as an experimental tool for troubleshooting, debottlenecking, and new product and process research and development.

The model for training operators and engineers and testing the control system must not be just a process model. It needs to be a plant model. It must include models of instrumentation and control valves (everything outside of the control system). Lags and delays should be added as needed to simulate the dynamics of sensors and actuators. It is important to simulate the lags of thermowells and electrodes and the stroking time of control or on-off valves greater than 4 inches since these are relatively slow. However, most of the other lags associated with sensors and actuators are negligible. More often, the dead time from the sensitivity and/or resolution limits of sensors, digital components, mechanical components, and control valves, the dead time from the cycle time of analyzers, and the dead time from transportation delays of temperature and composition changes in pipelines are the limiting factors to control system and thus plant performance. The resolution limit of control valves and mechanical components can be simply simulated by a dead band (backlash) and the resolution limit of sensors and digital components by a quantizer. The dead time from cycle times can be simulated by a various types of sample and holds. The input (read) and the output (write) are done at the beginning of the cycle for digital devices, whereas the input is done at the beginning and the output is done at the end of the cycle for analyzers. The dead time of transportation delays for pipelines and ducts is the volume divided by the flow rate. The dead time of transportation delays for spin, conveyor, and sheet lines is the length divided by the speed.

The effect of noise is also significant and can be more of a problem than dead time. All measurements have noise (even temperature sensors); it is just a matter of how much and how slow. The control system can distinguish a load upset only if it is larger than the noise amplitude. If the noise frequency approaches the control system natural frequency, the noise is amplified. Also, if the control valve moves as the result of noise, the control system inflicts a disturbance on itself.

If the plant model has a friendly graphical user interface (GUI) and high-level functions and offers presentations suitable for design and documentation, as shown in Fig. 1, it can become the source and the depository of plant knowledge that is readily maintained and widely used. The same principles are true for the control system. The common control language established by the Fieldbus standard (see section 11) and used in field-based systems (see section 3) offers this capability. If these plant models and configurations can be interfaced and combined in a personal computer, the plant and control system knowledge is centered in the training system and the paradigm for operator training is shattered.

The major architectures used in computer-based operator training systems involve major distinctions as to modification, location, and interconnection of the major components (plant model, control system, trainer interface, and operator interface). Often these reside in different personal computers to allow the trainer, model support person, and trainees to have simultaneous and separate access to the training system, as shown in Fig. 2. When the operator interface uses special computers and screens and the control system can reside in only a proprietary microprocessor-based controller, the connection between the controller inputs and outputs and the model is done by special communication packages and serial interface cards. When the control system and/or interface can be emulated in



8.35 **FIGURE 1** An operator training model that serves as the source and depository of plant knowledge. (© Hyprotech Ltd. 1998.)

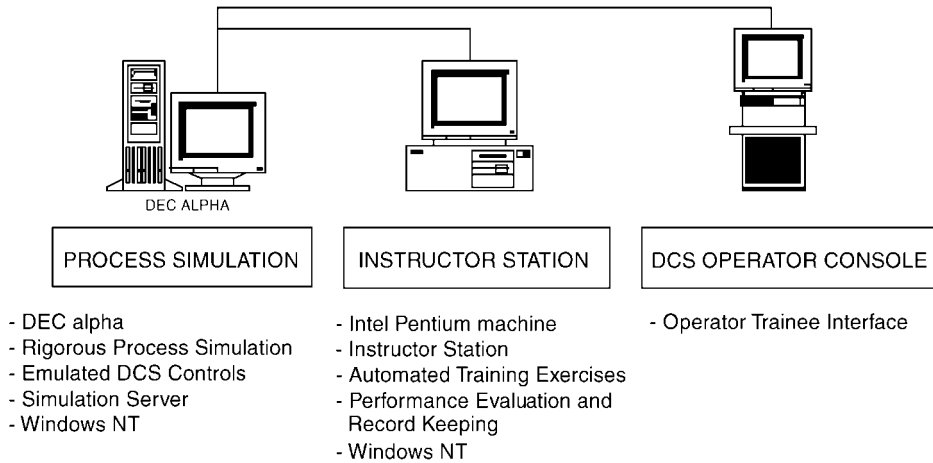


FIGURE 2 An operator training system network with separate stations. (© Hyprotech Inc., 1998.)

the same computer as the plant model, there is an opportunity to develop, test, and demonstrate the model and control system on a personal computer. The advantages of emulated control systems in terms of being able to implement and use the trainer anywhere and share results with everyone cannot be overemphasized. The control system emulation has additional value in terms of opening up the possibility of running the control system faster than real time and inherently providing initialization of setpoints and outputs. If the control system for the faster than real time studies is distinct from the DCS or the PLC control system, there are extensive duplication and initialization requirements. However, the emulation of proprietary operator screens and keyboards as soft keys or new keys on a PC is undesirable since it fails to ensure that the operator can learn to quickly make changes during real operating crises. As DCSs and PLCs develop true Windows interfaces, the changes can be transparent to the operator and the emulation of the interface becomes both feasible and highly desirable. The Windows NT environment and object linking embedding for process control enables the communication between the simulation, control system, and operator interface without special software or hardware and enables the system to run completely on a personal computer.

The connection between the plant model and the DCS or the PLC must be at the input and the output blocks of the configuration. If the connection is done to the loop via the data highway, signal calculations (split ranging and compensation), characterization, and selection are bypassed. The configuration and the performance of these functions are generally more of an issue than the tuning settings of the loop. It is critical that the training system include all of the input and output signal processing in the configuration.

REFERENCES

1. Lauritsen, M., "Dynamic Simulation Applications and Benefits to Engineering and Operations," presented at the American Institute of Chemical Engineers CAST meeting, September, 1998.
2. McMillan, G., *Modeling and simulation of processes, in Instrument Engineer's Handbook*, 3rd ed., Chilton, Location, 1994, pp. 70–76.
3. HYSYS™ is a trademark of Hyprotech Ltd. (information and figures are reproduced by permission of Hyprotech Ltd.).

INTELLIGENT ALARMS

G. K. McMillan*

Smart alarms or the equivalent alarm filtering can eliminate 90% or more of the alarm presented to the operator [1]. The goal should be zero alarms for normal operation and a single alarm for an abnormal condition that pinpoints the root cause. This is especially important during upset conditions because the response of the operator rapidly deteriorates as the number of alarms increases. Figure 1 shows that the operator needs at least 1 min after one alarm, 10 min after two alarm, and 20 min after three alarms to have any chance at all of a proper diagnosis as to the cause of the alarm so that the probability of failure to diagnose the alarm is less than one. Figure 1 also shows that the operator needs 10 or more minutes to ensure that the probability of failure to diagnose a single alarm is low (<0.1) [2].

Instead of triggering alarms off of a high or low measurement, the alarm should be built up to show the actual operating condition from information from diverse sources such as sensors, tasks, modes, outputs, and other alarms. If done properly, a single alarm is generated when the root cause is identified. Alarm filtering can be made to accomplish this same goal after the fact by use of a similar logic to present to the operator only one alarm related to the root cause. However, the number of alarms generated is very large and the highway traffic is very high during critical periods. Thus it is better to build in the proper logic in the beginning to reduce the number of alarms generated rather than filters to screen out the extraneous alarms. Alarm analyzers that tell the operator which

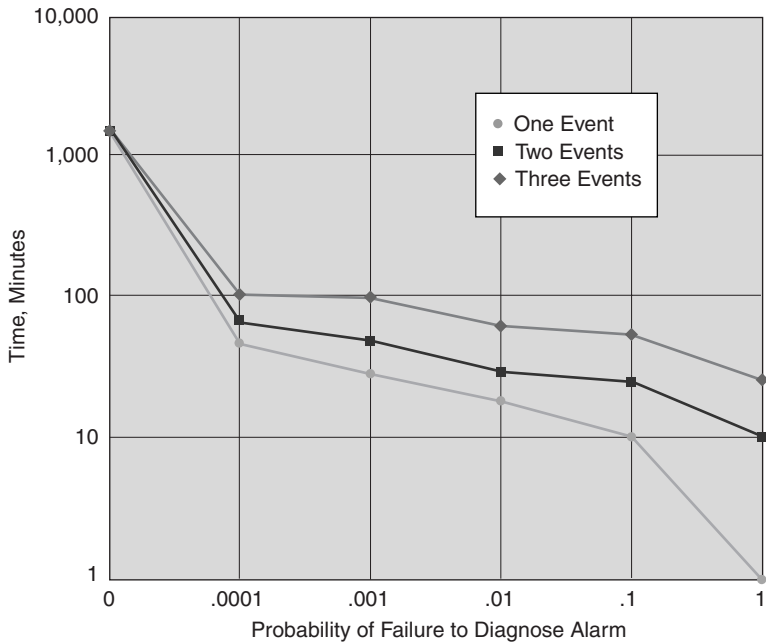


FIGURE 1

* Senior Fellow, Solutia Inc., St. Louis, Missouri.

one of the alarms was really important are too late because the damage has already been done by the barrage of alarms. Even if the extraneous alarms do not activate a horn or flash, they still pose a serious undesirable and unnecessary distraction to the operator. Thus, to minimize highway loading and operator distraction from extraneous information, the control systems should reduce the number of alarms generated rather than screen or classify alarms.

Each of the conditions that set a smart alarm needs to have the option of a time delay to ignore spurious signals and allow time for a valves, pumps, process, and instruments to respond. Also, the conditions to set an alarm should be based on analog and discrete measurements (sensors) so that the failure of motors or valves does not cause a false alarm. On the other hand, conditions to clear the alarm should be generally built up from action requests (set points for unit points, positioners, and discrete control devices) rather than feedback measurements or statuses to ensure conditions to deactivate the alarm are recognized early enough. Also, each analog value used to set and clear an alarm must have a dead band to eliminate alarm chatter from noise.

The basic features for build a simple smart alarm are illustrated by the following logic used to activate a smart alarm to alert an operator that a batch has gotten too thick (e.g., excessive polymerization). In this example, there is an agitator and a thickening task in the batch sequence.

The conditions to set a smart alarm to alert the operator that the batch is too thick are as follows:

1. The batch sequence unit point is in thickening task for more than 5 s.
2. The agitator amps are too high for more than 5 s.

The conditions to clear a smart alarm that indicates that the batch is too thick are as follows:

1. The batch sequence unit point requested task is anything but thickening.
2. The agitator amps have dropped below the alarm point minus the dead band.

The basic features required for building a more complex smart alarm are illustrated by the following logic used to activate a smart alarm to alert the operator that a feed nozzle or control valve is plugged to a fed batch reactor. In this example, consider that there is a feed tank, pump, solids filter, flow control loop, automated isolation valve, smart alarm to notify the operator that the feed filter is plugged, and a batch sequence. The smart alarm for the plugged filter might use pressure drop and would be considered to be a precursor to the low flow condition for the smart alarm for the plugged nozzle or control valve.

The conditions to set a smart alarm to alert the operator that a feed nozzle or control valve is plugged are as follows:

1. The feed tank level measurement is above 4% for more than 14 s.
2. The batch sequence unit point is in the feed task for more than 10 s.
3. The pump motor run contact is closed for more than 8 s.
4. The actual valve position from the smart positioner is greater than 40% for more than 6 s.
5. The isolation valve open limit switch is activated for more than 4 s.
6. The smart alarm that indicates that the feed filter is plugged is not activated for more than 2 s.
7. The flow measurement field status is good for more than 10 s.
8. The flow measurement is less than 10% for more than 4 s.

The conditions to clear the smart alarm that indicates that the feed nozzle or control valve is plugged are as follows:

1. The feed tank level measurement is below 2%.
2. The batch sequence unit point requested task is anything but the feed task.
3. The pump motor set point is stopped.
4. The flow controller output (smart positioner set point) is less than 20%.

5. The isolation valve set point is closed.
6. The smart alarm that the feed filter is plugged is activated.
7. The flow measurement field status is bad.
8. The flow measurement is above 20%.

In the above examples and for most applications, one can derive the conditions to clear the smart alarm from the conditions to set the alarm by substituting set points for measurements, by using dead bands for analog values, and by omitting time delays. However, for the more general case, it may be advantageous to offer the user the option of deriving the clear condition from the set condition and adding distinct or separate clear conditions. In other words, for each set condition, the user would fill in the tag of the loop, indicator, discrete control device, unit point, or precursor alarm followed by the associated trigger point, time delay, and dead band, and finally check a box as to whether to automatically derive the clear condition. Finally, an optional entry of specifically different clear conditions would be offered.

The alarm is activated (st) or deactivated (cleared) or bypassed by the operator by means of a discrete status rather than by changing trip points between designated and unreachable values. Newer DCS systems support this feature, which eliminates the need for housekeeping of valid and invalid analog alarm settings.

The above method of constructing smart alarms leads the user into more carefully thinking of the conditions that uniquely identify the abnormal operating condition. In contrast, most DCSs ask the user to fill in the high, high-high, low, and/or low-low settings for each analog point and to choose a deviation alarm for all points. In early DCS designs, a setting was mandatory and the only way to deactivate an alarm was to choose a setting that could never be reached. As a result, the DCS created an order of magnitude increase in alarms compared with panel board analog controllers. The DCS now has the opportunity to generate more meaningful and fewer alarms than those of the analog days by giving the user the ability to go exclusively to smart alarms. Additionally, the advent of the field-based DCS opens the ability to use diagnostics from smart instrumentation to help make alarms smarter.

REFERENCES

1. Bray, M., and D. Corsberg, "Practical alarm filtering," *InTech*, February, 34–36, 1994.
2. Heavner, L., L. Schultze, and G. McMillan, "Dynamic alarming for DCS," *InTech*, November, 22–25, 1993.